THE BENEFITS OF ACOUSTIC PERCEPTUAL INFORMATION FOR SPEECH PROCESSING SYSTEMS

BY

DI HE

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2019

Urbana, Illinois

Doctoral Committee:

Professor Deming Chen, Chair
Professor Mark Hasegawa-Johnson
Professor Martin Wong
Boon Pang Lim, Novumind, Inc.

Abstract

The frame-synchronized framework has dominated many speech processing systems, such as ASR and AED targeting human speech activities. These systems have little consideration for the science behind speech and treat the task as a simple statistical classification. The framework also assumes each feature vector to be equally important to the task. However, through some preliminary experiments, this study has found evidence that some concepts defined in speech perception theories such as auditory roughness and acoustic landmarks can act as heuristics to these systems and benefit them in multiple ways. Findings of acoustic landmarks hint that the idea of treating each frame equally might not be optimal. In some cases, landmark information can improve system accuracy through highlighting the more significant frames, or improve the acoustic model accuracy by training through MTL. Further investigation into the topic found experimental evidence suggesting that acoustic landmark information can also benefit end-to-end acoustic models trained through CTC loss. With the help of acoustic landmarks, CTC models can converge with less training data and achieve lower error rate. For the first time, positive results were collected on a mid-size ASR corpus (WSJ) for acoustic landmarks. The results indicate that audio perception information can benefit a broad range of audio processing systems.

To my Wife, Father and Mother.

Acknowledgments

This dissertation would not exist without the support of many people. Many thanks to my adviser, Deming Chen, who read my numerous revisions and helped make some sense of the confusion. Also thanks to my co-advisor and committee members, Mark Hasegawa-Johnson, research collaborator and committee member Boon Pang Lim, and co-advisor and committee member Martin Wong, who offered guidance and support. Many thanks to Xuesong Yang, Zuofu Cheng, Dushyant Bhan, Zizhen Liu, Xinheng Liu, Swathi Gurumani and Kyle Rupnow for collaborating and contributing to the research. I would like to thank Hui Jiang (who also happens to be my wife) for contributing to data annotation. Some collaborators are partially funded by Qatar National Research Fund (QNRF) grant 7-766-1-140 and the DARPA LORELEI program. All results and conclusions are those of the author and collaborators, and are not endorsed by DARPA. And finally, thanks to my wife, parents, and numerous friends who endured this long process with me, always offering support and love.

Table of Contents

List of Abbreviations . . . . . **vii**

Chapter 1 Introduction **1**

Chapter 2 Using Auditory Roughness as Pre-filtering Feature for Human Screaming and Affective Speech AED **7**
2.1 Background 9
2.1.1 Distributed AED 9
2.1.2 Auditory Roughness 10
2.2 Method 11
2.2.1 Building the Front-end of a Distributed AED on FPGA 11
2.2.2 Acting as a Pre-filtering Front-end 14
2.2.3 Approximating Auditory Roughness 17
2.3 Results 19
2.3.1 Performance of the AED 19
2.3.2 Mandarin Affective Speech 20
2.3.3 Youtube AudioSet 22
2.3.4 Beyond Pre-filtering 27
2.4 Recapitulation 27

Chapter 3 Using Acoustic Landmarks to Improve ASR through Frame Dropping and Frame Re-weighting **29**
3.1 Background and Literature Review 31
3.2 Measures of the Information Content of Acoustic Frames 36
3.2.1 Frame Re-weighting 37
3.2.2 Frame Dropping 38
3.3 Hypotheses 40
3.4 Experimental Methods 41
3.5 Experimental Results 42
3.5.1 Hypothesis 1: Over-weighting Landmark Frames 43
3.5.2 Methods of Replacement for Dropped Frames 43
3.5.3 Hypothesis 2: Dropping Frames with Regard to Landmarks 47
3.6 Discussion 52
3.6.1 How Landmarks Affect the Decoding Results 53
3.7 Recapitulation 57

Chapter 4 Using Acoustic Landmarks to Improve ASR through MTL . 61
4.1 Background . . . 63
4.1.1 Multi-task Learning . . . 63
4.1.2 The Iban Corpus . . . 64
4.2 Methods . . . 65
4.2.1 Defining and Marking Landmarks . . . 65
4.2.2 Adjusting Landmark Labeling . . . 66
4.2.3 Cascading the MTL to Iban . . . 67
4.3 Results . . . 69
4.4 Recapitulation . . . 71

Chapter 5 Using Acoustic Landmarks to Improve CTC Training through Label Sequence Augmenting . . . 72
5.1 Background . . . 73
5.1.1 Connectionist Temporal Classification (CTC) . . . 73
5.1.2 Acoustic Landmarks . . . 75
5.2 Methods . . . 75
5.2.1 Distinctive Features and Landmark Definition . . . 75
5.2.2 Augmenting Phone Sequences with Landmarks . . . 76
5.2.3 Acoustic Modeling using CTC . . . 78
5.3 Experiments . . . 78
5.3.1 Configurations . . . 78
5.3.2 Experiments on TIMIT . . . 80
5.3.3 Datasets Smaller and Larger than TIMIT . . . 81
5.4 Recapitulation . . . 83

Chapter 6 Conclusion and Future Work . . . 85
6.1 Summary of Key Contributions . . . 85
6.1.1 List of Contributions . . . 85
6.1.2 Connecting the Findings . . . 87
6.2 Future Work . . . 87
6.2.1 A Better Acoustic Landmark Detector . . . 88
6.2.2 Re-defining Acoustic Landmarks . . . 92
6.2.3 Augment CTC Training through Other Means . . . 93
6.2.4 Acoustic Landmark and Attention Models . . . 94
6.2.5 Verify the Findings on Larger Corpus . . . 95

References . . . 96

List of Abbreviations

AED	Audio Event Detection
AM	Acoustic Model
ANN	Artificial Neural Network
AR	Auditory Roughness
ASR	Automatic Speech Recognition
CE	Cross Entropy
CMAC	Complex Multiply Add Accumulate
CTC	Connextionist Temporal Classification
CV	Cross Validation
DBN	Deep Belief Network
DCT	Discrete Cosine Transform
DNN	Deep Neural Network
DSP	Digital Signal Processing
FBank	Filter-bank
FIR	Finite Impulse Response
FMLLR	Feature space Maximum Likelihood Linear Regression
FP	False Positive
FPGA	Field Programmable Gate Array
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model

IIR	Infinite Impulse Response
IoT	Internet-of-Things
IPA	International Phonetic Alphabet
LSTM	Long-short Term Memory
LVCSR	Large Vocabulary Continuous Speech Recognition
MAC	Multiply Add Accumulate
MFCC	Mel-frequency Cepstral Coefficient
MTL	Multi-task Learning
PER	Phone Error Rate
ROC	Receiver Operating Characteristic
STM	Short-term Energy
SVM	Support Vector Machine
SWM	Subtract Windowed-mean
TDNN	Time-delayed Neural Network
TP	True Positive
UART	Universal Asynchronous Receiver-transmitter
WER	Word Error Rate
WFST	Weighted Finite State Transducer
ZCR	Zero-crossing Rate

Chapter 1

Introduction

Speech processing systems have come a long way from machines with tight limitations on vocabulary size, recording channel and noise level to the robust, portable and accurate systems they are today. Recent studies on neural networks have resulted in significant performance gains for modern ASR systems [1, 2] and speech-related AED systems [3]. Under clean recording conditions, the accuracies of some of these systems are approaching human level.

In some scenarios, some speech processing tasks have been considered to be solved problems or engineering questions. However, in most cases, the current state-of-the-art performance is still far from ideal. Take ASR for example: the systems [1, 2] that demonstrate close to human performance accuracy need to operate in very restricted conditions. Whenever there is noise, room reverberation or other channel effect present, machine performance drops dramatically below what can be achieved by a human. What is more, there is the case of multiple speakers. The classic cocktail party problem [4] is still a very difficult research question today. Yet, despite many efforts, the machine performance in this scenario is still not remotely close to human performance. There still seems to be audio perception ability common to humans that machines cannot not master. It is fair to claim that speech-processing researchers still face many meaningful challenges today. Active research findings appear rapidly and people benefit from their contributions.

Since the tri-phone acoustic model was introduced [5], a considerable amount of time ago, the GMM-HMM framework has dominated the ASR area. GMM was also applied to many other speech processing systems, such as AED targeting human vocal events [6].

Many of these systems have one key characteristic in common: they embrace a frame-synchronized methodology. Under this methodology, speech features have been extracted at a fixed rate and all extracted feature vectors will be scored against the pre-trained acoustic model. For a subset of these systems that come with a temporal model, usually in the form of a HMM, more informative feature vector frames might contribute to the decoding process more significantly. However, these systems always assume each feature vector to be equally important. This assumption is common for statistic model handling sequential input, where no heuristics on the input is available. In recent years, abundant training data and more powerful computing platforms contributed to the rise of deep neural network models [7]. The GMM acoustic model has been replaced by DNN and even recurrent models such as LSTM.

This work intends to find out if there exists useful information defined in speech perception theories that can augment audio processing systems. The audio processing systems under investigation are ASR and AED targeting speech event such as screaming and affective speech. Needless to say, both applications have significant impact on people's daily lives. Improving their performance will benefit the human society undoubtedly. The audio perception theories mentioned above are theories defined in audio perception and articulation science. These studies include articulatory phonetics, the study of how the speech sound is produced through the interaction of different physiological structure, illustrated in Fig 1.1. They also include acoustic perception, which is how different characteristics in sound affect the way we perceive them. We will cover more details on the theory in the background section in each respective chapter.

The natural followup question that calls for an answer is why audio perception information is chosen over other measures. Further, why do audio processing systems bother with extra information in addition to classic MFCC or Fbank features? The second problem can be easily answered. Take ASR for example, where speaker identity, gender and a lot of

[0] https://en.wikipedia.org/wiki/File:Illu01_head_neck.jpg

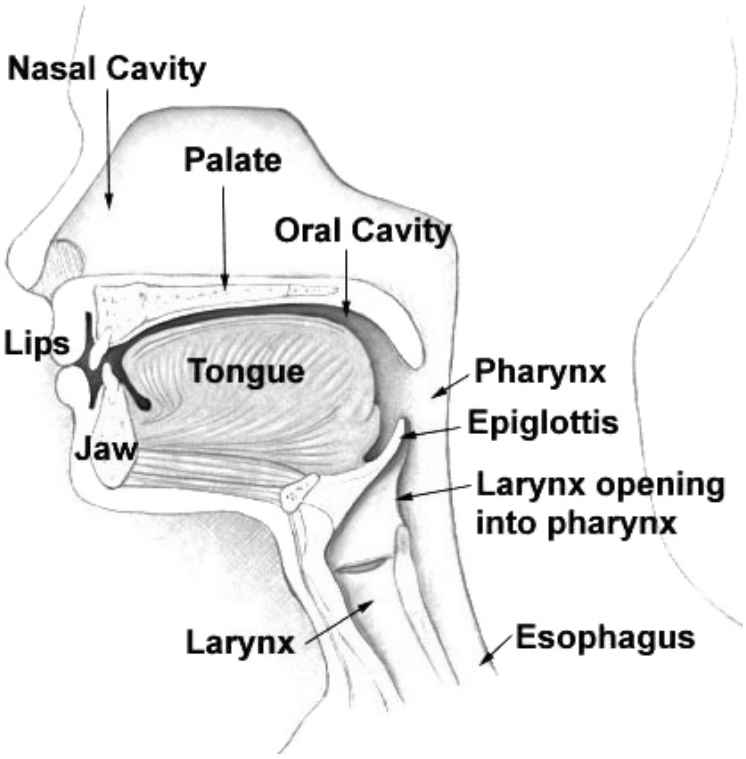


Figure 1.1: Human vocal tract.

other speech attributes contributed heavily to improving ASR accuracy. Without the help of different kinds of speaker adaptation (FMLLR [8] and I-Vector [9]), ASR systems cannot reach the accuracy level they have achieved today. The answer to the first question will be more complex and depend on the actual perception information under discussion. The main purpose of this study will be to provide theory and experimental support to their inclusion in audio processing. However, in general, audio perception and articulation theories approach the audio process from different angles. Many of the measures and cues introduced in these theories are salient to human perception. Consider the classic MFCC and Fbank features, for example; they are also based on human perception, and there is a good chance that this audio perception information contributes positively.

Since many cues and measurements defined in audio perception have outstanding instantaneous signature on the power spectrum, it is more intuitive to assume this information benefits frame synchronized audio processing systems over end-to-end systems. End-to-end

systems are usually trained on more abstract label and loss criteria. As a result, the connection between output labels and audio characteristics is less direct. As we will see in the background parts of Chapters 3 and 4, many previous works attempt to augment audio processing, specifically speech processing systems, with speech perception information. These works, without exception, focus on frame-synchronized systems.

While frame-synchronized statistical models still play a key role and are dominant in speech processing systems, systems that do not fall in the same framework are also gaining attention. This is because frame-synchronized systems, compared to end-to-end systems, have shortcomings. First of all, these systems tend to be more complex in terms of the kind of computations conducted. Since audio processing systems, especially speech processing systems, tend to have fairly high frame rate, a single frame usually is too short to last a full acoustic unit or event. As a result these systems tend require a temporal model in parallel to an acoustic model. The type of computations used to infer the acoustic models and temporal model tend to be dramatically different. Secondly, training an acoustic model for these systems requires state alignment information. This alignment information usually comes from inferring another acoustic model. As a result, training an accurate acoustic model usually requires a number of iterations. In addition, since the frame rate of frame-synchronized systems tends to be high, the acoustic units used by these systems tend to be much more fine-grained than the audio units that are familiar to human understanding. Take speech for example: these systems usually operate on tri-phone states, instead of grapheme or phoneme. Compared to 24 letters or around 40 to 60 phones in English, a LVCSR system in the same language tends to have a couple thousand tri-phone states. This introduces a significant size burden on the temporal model, making it hard to efficiently store and slow to infer.

To address these issues, audio processing systems based on an end-to-end framework were introduced. Among these end-to-end systems, there are systems based on connectionist temporal classification (CTC) models [10] and systems based on attention models [11].

Although there is still no clear indication that these end-to-end systems are fully replacing classic frame-synchronized systems yet, these systems [12, 13, 14] are already gaining more and more attention. However, for these models, very little abstract audio information can be extracted directly, and so trying to extend the use of audio perception information to these systems can be very difficult. Frankly, no previous work has successfully demonstrated a successful example.

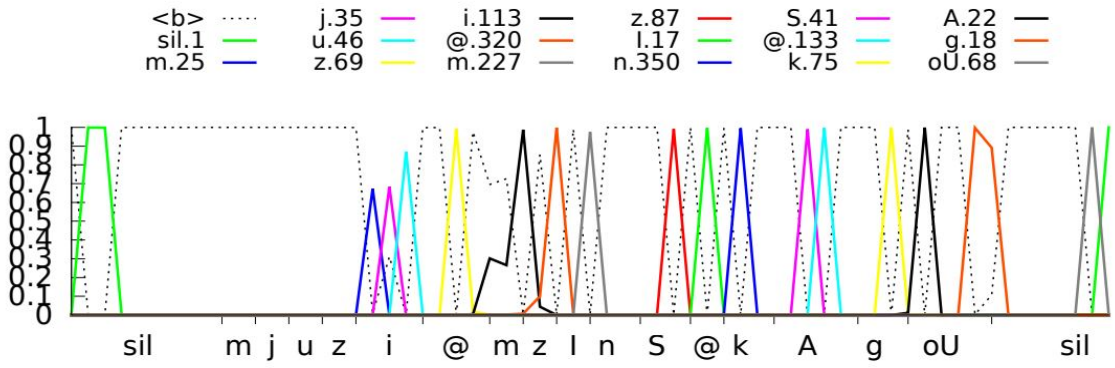


Figure 1.2: Unidirectional context-dependent CTC output.

Take the CTC models for ASR for example. Figure 1.2 from [10] illustrates a classic output of a CTC model. CTC defines loss based on mismatch in the output and target phone sequence as opposed to `hard` frame-by-frame difference between frame-wise output and label. Therefore, the training procedure does not penalize phone labels generated outside of the phone duration, as far as it is part of a correct sequence. This results in output similar to that presented in Fig 1.2. As we can see, prediction of many phones, `'m'` for example, appeared long after the pronunciation of `'m'` ended.

Heuristics are deemed beneficial to speech processing systems if applying them to the system results in either improvement in the recognition or detection accuracy or a reduction in the computational complexity while maintaining similar accuracy. Due to the rapid increase in computational capability, especially as cloud computing becomes more and more practical and reliable, many new models have traded heavier computational load for higher accuracy; however, porting mobile platforms and the framework of IoT still pose a serious restriction

on computational load. Therefore, if for the same or similar accuracy, computational load can be reduced, speech applications still benefit.

Partnering with collaborators, we have examined multiple speech perception concepts and found two of them promising. Chapter 2 presents experimental findings supporting auditory roughness [15, 16, 17] as a pre-filtering feature for screaming and affective speech AED. We found that with computational complexity comparable to that of STE, auditory roughness outperforms STE, or other features with similar complexity, when used to detect screaming or affective speech. Chapter 3 presents preliminary results on augmenting the ASR acoustic model scoring process with acoustic landmark [18, 19] information. Experimental results tend to confirm that over-weighting feature frames containing acoustic landmarks reduces ASR PER. In addition, dropping frames not containing acoustic landmarks can dramatically reduce acoustic model scoring computational load at the expense of relatively small accuracy loss. In Chapter 4, acoustic landmark information is used to improve ASR DNN acoustic model through MTL [20, 21, 22]. Experimental results were encouraging on an English corpus. In addition, landmark detector trained in English was also able to reduce WER on corpus in a different language. The strongest contribution of this work is explained in Chapter 5, where acoustic landmark information is employed for the first time in an end-to-end system and experimental results show that this information can improve CTC AM accuracy and improve model convergence ability. To the best of our knowledge, this is the first work to apply acoustic landmarks to an end-to-end ASR system; it is also the first work to conduct experiments with acoustic landmarks on mid-size ASR corpus.

To allow for a compact illustration of each study, each chapter contains background, methodology, experimental results and recapitulation section specific to the study. Unlike the work on auditory roughness, where the study can be concluded, the study of applying acoustic landmarks to ASR is still open. We conclude the thesis and talk about potential future directions to advance the study in Chapter 6.

Chapter 2

Using Auditory Roughness as Pre-filtering Feature for Human Screaming and Affective Speech AED

Internet of Things (IoT) has provided a new approach to many applications. When powerful and reliable computational capability meets distributed wireless sensor networks, many tasks that used to suffer from cost, practicality and low-accuracy benefit [23]. Audio event detection (AED) based security or surveillance systems are one of these applications.

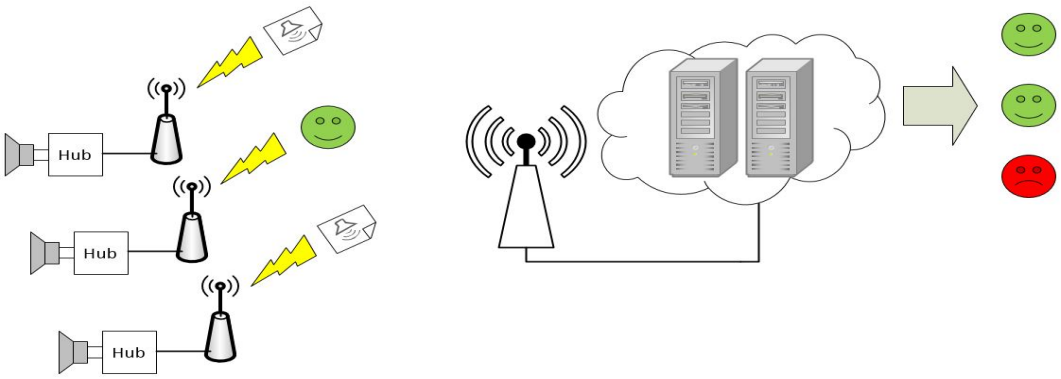


Figure 2.1: Distributing AED across the network.

AED for security purposes has been an interesting topic to both the research community [24, 6, 25, 26] and to commercial entities [27, 28]. Security events such as gunshots and explosions are relatively easy to detect, and commercial products based on AED have already been deployed in many US cities [27, 28]. Security related human speech events, such as screaming, shouting, and other manifestations of fear and anger have proven to be more difficult to detect accurately; many works train dedicated models to detect these speech events [6, 25]. State-of-the-art methods include deep neural networks (DNN) and hidden markov models (HMM) [29] and long short-term memory recurrent neural networks (LSTM RNN) [30]. The computational complexity of these methods is high. The IoT ap-

proach, presented in Fig 2.1, does offer AED access to powerful computation capability. However, having enough power dedicated to each and every sensor all the time is inefficient and impractical for large sensor networks. AED systems targeting human speech therefore use *pre-filtering* mechanisms: algorithms with low computational complexity that can detect events of interest with a low missed detection rate, and with a false alarm rate that may be relatively high, but that is low enough to limit the computation expended by the second-pass classifier [24, 25, 31]. Pre-filtering can reduce communication and computation costs by discarding audios when an event is likely absent.

Previously published pre-filtering algorithms fail to meet the three simultaneous requirements of high recall, acceptable precision, and extremely low complexity. The problem usually is the intrinsic limitation of the *features* used for *pre-filtering*. Some of these works [24, 31] rely on windowed short-term energy (STE). STE, although light in computation, as we will show later, fails to differentiate affective speech and neutral speech effectively. Other work [25] uses spectral features, which are much complex, on the order of over 10×, to extract than STE [32].

This chapter considers the potential of a classical acoustic feature called the auditory roughness [15, 16, 17], as a pre-filter feature. Auditory roughness is a classic measure of "harsh and unpleasant" sound with a long history. Although it used to be an acoustic concept, recent biological studies have found evidence that fear is triggered in the mind by perceiving human screaming [17]. First, the standard auditory roughness feature is used to detect anger and fear in the Mandarin Affective Speech corpus [33], and screaming in the Youtube AudioSet [34], with recall and precision better than STE. Second, since the standard roughness computation has complexity similar to that of spectral features, an approximate roughness measure is proposed, with computational complexity similar to STE and at least 10× lighter than MFCC. The approximated auditory roughness feature is demonstrated to have recall and precision better than other pre-filtering features with similar computational complexity, including STE.

We will briefly introduce the concept of distributed AED and auditory roughness in Sec 2.1. We will explain how and why we approximate roughness in Sec 2.2. Experiment setup and results are then presented in Sec 2.3. We conclude and discuss the work at the end.

2.1 Background

2.1.1 Distributed AED

A distributed surveillance system designed under the framework of [23] would look like Fig 2.1. A large number of sensing nodes with wireless capability are deployed. Different kinds of sensors are grouped into each node and correlated by a hub controller. The controller must run non-stop to serve the sensors conducting minimum surveillance; it must also power-on and control more powerful sensors when needed. If pre-filter operations find signs of an event happening, features collected by the sensors will be uploaded to the cloud for further analysis.

In an AED scenario, if pre-filtering features do not exceed normal level, the audio is only buffered for a couple of seconds. Hubs will inform the cloud that everything is fine. If the pre-filtering feature exceeds a preset threshold, a request is initialized: the buffered audio is packed and uploaded for further analysis. Depending on the connection, further audio is either packed into chunks and uploaded, or streamed to the cloud as the event unfolds. On the cloud side, more powerful servers can be accessed on demand. If a request comes in, the servers will run much more sophisticated algorithms such as DNN+HMM or LSTM to conduct more complicated analysis. To build and run a system with a large number of sensors, both the sensors and the controllers must be cheap and low-power. Hardware targeting sensor hubs have limited resources, e.g., they typically have no dedicated multiplier.

2.1.2 Auditory Roughness

The term "auditory roughness" originated as a musical expression in the 19th century [15]. The term is now defined to be a psychophysical dimension, describing the human perception of harsh, raspy hoarse sounds [16]. The musical and perceptual concept was formalized as a sound quality measurement, with several standard definitions and published algorithms. [1] Recent studies have identified apparent neurobiological correlates of perceived auditory roughness [35].

Amplitude modulation frequency is one of the most important physical acoustic correlates of auditory roughness [36, 17]. In music and other non-vocal sounds, a modulation frequency of 30Hz or below is usually perceived as beats [36]. When modulation frequency exceeds 30Hz, it is considered rapid and the sensation of roughness appears. Though speech is complex, the same modulation frequency thresholds seem to apply: neutral speech has most of its modulation spectral energy at 1–10Hz [37], and modulation frequencies above 30Hz will trigger the brain's fear center [35]. The sense of roughness peaks at modulation frequencies of 70Hz [17] or 75Hz [36], but persists in response to modulations of up to 150Hz [36] or 300Hz [17].

Further studies [36] claim carrier frequency and the strength of amplitude modulation also affect the level of roughness. Some auditory roughness calculators [38, 36] consider each spectral peak as a carrier frequency, compute the modulation frequency and modulation strength of each carrier, and add the roughness effects from each carrier frequency together. Other algorithms compute auditory roughness by analyzing the distribution of energy within pre-determined frequency bands, e.g., in 24 bands uniformly distributed on a Bark scale [39, 17]. [1]

[1] `http://www.ni.com/product-documentation/8169/en/#toc4`

2.2 Method

2.2.1 Building the Front-end of a Distributed AED on FPGA

Overall IoT System Design

A block diagram of our design is found in Fig 2.2. For the purposes of the DAC-IoT competition, we have designed and manufactured a new PCB board, which consists of four microphones, associated amplifiers, and I2S analog to digital converters. This board interfaces to the Lattice XO3 Starter Kit through a pin-header. The XO3 FPGA receives the digitalized audio, normalizes the amplitude, and extracts 4 DSP features. These features consist of: short-term energy (STE), subtracting windowed median (SWM), zero-crossing rate (ZCR), and auditory roughness (AR). The FPGA buffers a window of 64 samples and computes the features across the window. The features are then fed into a small classifier module on the FPGA, which detects and classifies the event type, while rejecting non-suspicious events. Suspicious audio events are combined with sound localization data, which provides an event direction using the microphone array. An encrypted packet with the classification and beamforming is transmitted wirelessly to a centralized hub using the FPGA's UART port. One centralized hub may control multiple FPGA IoT edge devices. If live video is desired, the centralized hub can control a camera and stream video of the event to a human operator after suspicious sound is detected. The human operator would make the eventual decision to activate an alarm or to notify authorities; such a human-in-the-loop strategy can significantly reduce false positives. Since our device can localize the sound, it can also guide the camera to mechanically turn to the direction of the suspicious sound, zoom in, and trace the event in real time. Our IoT device can also be used for other purposes. For example, the centralized hub would activate the recording from the camera only when a suspicious sound is detected, thus both saving power and avoiding privacy concerns. Nonetheless, given the limited time and the focus of the FPGA-based IoT design, the camera part is beyond the

scope of the scope of this discussion.

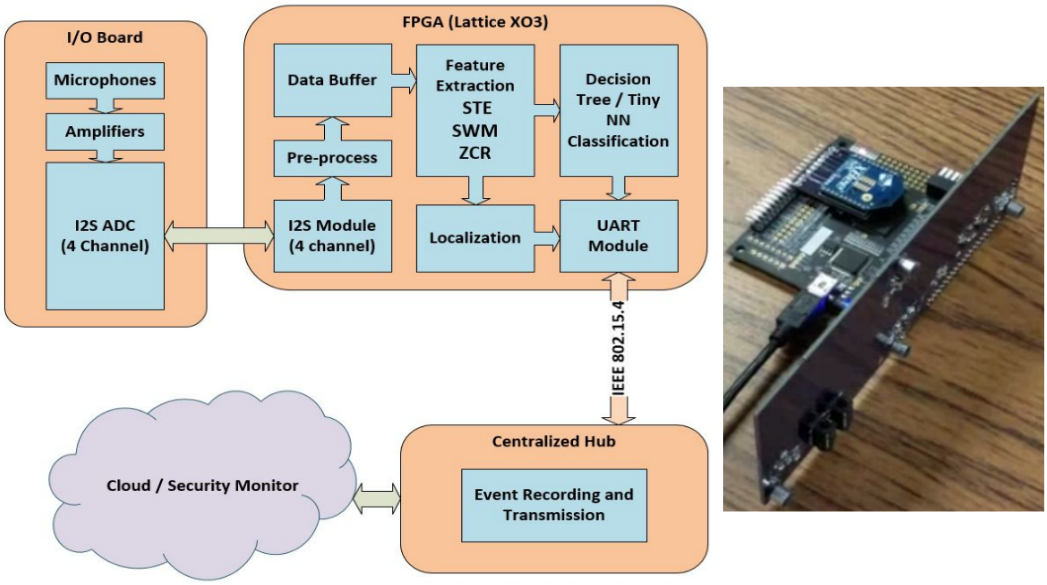


Figure 2.2: The AED system.

Our demonstration focuses on the suspicious sound recognition and localization. The novelty of our design includes: 1) There is no existing IoT device that detects screaming for security surveillance purposes using FPGAs. 2) We study various types of light-weight classifiers, including decision tree, neural network, and LSTM (long-short term memory), and carry out a thorough design space exploration. The final design is optimized for the best resource-accuracy trade-off. 3) The four features developed (STE, SWM, ZCR, and AR) can be used as IPs and be selected with different combinations targeting different accuracy levels for different FPGAs. 4) Our way of implementing beamforming is unique and novel, which can deliver high accuracy with very limited resource.

Signal Processing on the FPGA

A significant challenge in our design is the use of the Lattice Mach XO3 FPGA as a sensor hub and DSP. While the XO3s flexible PLLs and high performance I/O make interfacing with the audio converters straightforward and the small FPGA is most ideal for low power

and low cost IoT designs, the low number of 4-LUTs (6900 in our provided board) combined with the lack of hardware DSP blocks do make the signal processing a great challenge. To explore the design space and determine the best way to efficiently use the limited FPGA resource to fulfill our ambitious goal, we implement four DSP features which require relatively low DSP complexity. The first feature is ZCR, which counts the number of sign changes of the audio signal within a fixed number of samples and therefore is strongly correlated to the fundamental frequency of the signal. The next feature is STE, which is the sum of signal power in each sample within a fixed number of samples. STE is very effective at loudness detection and is widely considered as one of the best computational-light feature for gunshot or screaming detection. However, we have found that by itself, STEs ability to discriminate between screaming and loud speech is weak, as loud speech and speaking near the recording device usually registers high STE independent of the speaker emotional status. Therefore, STE must be used in conjunction with other features. The next feature is SWM, which computes the median for a sliding window around each sample and subtracts the sample with the windowed median. Since computing median involves sorting and can be computationally expensive, we approximated the feature with windowed mean and obtained similar results. Compared to all other features SWM is extracted at a sample rate, rather than the classifier rate of about 125 Hz. Therefore, we take the maximum within a fixed amount of samples to obtain the SWM at the classifier rate. Finally, the feature AR is used to be a psychophysical dimension, describing the human perception of harsh, raspy hoarse sounds and can be considered an industry standard for unpleasant sound level measurement. The standard procedure of calculating AR involves calculating the amplitude modulation rate, frequency and magnitude modulation between mono-frequencies within the audio. We approximated the calculation by looking at the signal amplitude around the key frequency (75Hz) of the audio amplitude. To effectively obtain the approximation, we use a short infinite impulse filter (IIR) to high-pass filter the modulated audio amplitude. Finally, we explored different combinations of these features targeting the best result for our IoT design

given the very tight resource limit on the FPGA.

2.2.2 Acting as a Pre-filtering Front-end

The motivation behind building such hardware was to demonstrate the practicality of the pre-filtering ideal. It has been proven that the proposed approximated is small enough and can be deployed on the FPGA mentioned above. However, we have to reduce the filter weights to 8-bit fixed point to accommodate for the resource limitation. The front-end board was given the ability to stream out pre-filter classification result and audio through a wireless communication link as shown in Fig 2.3 and Fig 2.4. The link is built based on a Zigbee module. To the machine on the receiving side, the pre-filtering results arrives from the serial port.

Figure 2.3: Transmitter for the AED front-end.

When building the demonstration, we also created a GUI to run a computer representing the centralized server. The GUI, as shown in Fig 2.5, is written based on Matlab and displays the pre-filtering alarms and the detected direction of the event.

A screen shot taken during actually running the device can be found in Fig 2.6. In the

Figure 2.4: Receiver for the AED front-end.

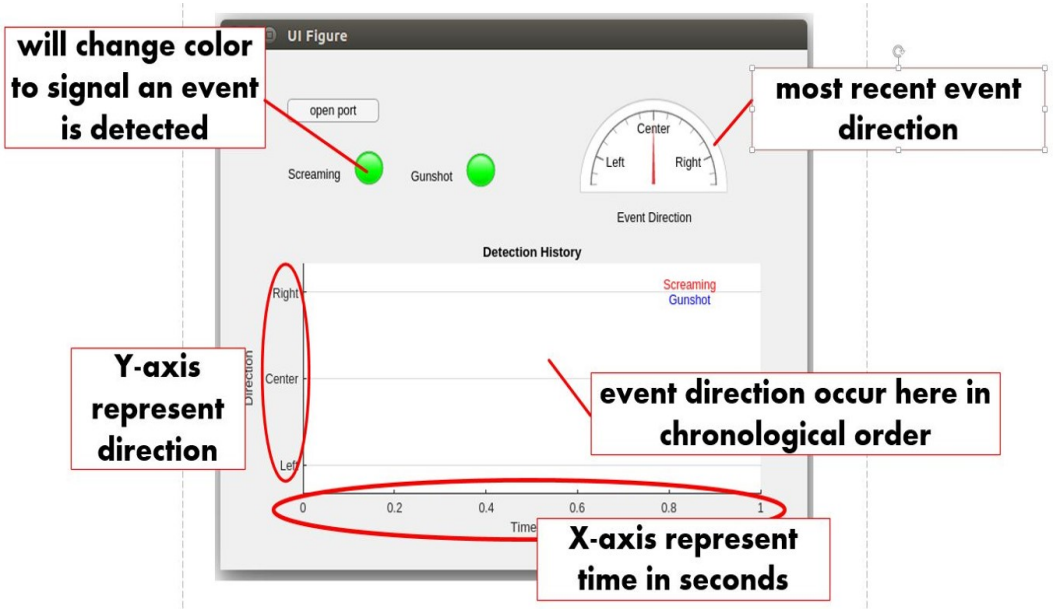


Figure 2.5: GUI for the AED on the host machine.

case shown in this figure, some screaming pre-filtering alarm was triggered on the left side of the front-end board.

Figure 2.6: GUI demonstrating the AED results.

Challenges, Optimizations, and Limitations

As stated previously, a major challenge was the implementation of real-time DSP as well as NN (neural network) based classification on a small FPGA without DSP blocks. We overcame this by careful selection of the audio features (none of our features required the use of a Fourier transform), as well as careful optimization of the Verilog RTL to timeshare expensive modules. For example, in the AR module, a single MAC module was shared in the implementation of the IIR filter. Wireless transmission was done using a UART module, to avoid the complexities of implementing a TCP stack on the FPGA. Our final resource usage is shown in Tab 2.1.

Due to some limitations of the FPGA, we were not able to simultaneously implement all the modules we designed on the Mach XO3-6900. For example, we found that implementing the ANN resulted in too much FPGA routing congestion unless the localization buffer size

Table 2.1: Resource utilization.

Number of Slices	Number of Block RAMs	Number of PLLs
2298 out of 3432 (67%)	7 out of 26 (27%)	2 out of 2 (100%)

was decreased, which would cause the localization accuracy to suffer. Use of a larger XO3-9400 would allow us to implement the entirety of the design simultaneously, improving classification and localization accuracy. With XO3-6900, our current implementation used the decision tree as the classifier. However, we believe our device demonstrates a useful application for the Lattice Mach XO3 in IoT applications.

2.2.3 Approximating Auditory Roughness

In our attempt to detect human screaming, fear, and anger, auditory roughness seems to be a good candidate feature. However, both of the standard methods used to compute auditory roughness are far too complicated for pre-filtering. We therefore propose a computationally much simpler approach to approximate roughness. Psycho-acoustic studies [35] agree that rapid and strong amplitude modulation, at around 30-150Hz, is the most important physical correlate of perceived roughness. Existing algorithms are computationally expensive because they assume that the speech signal contains many different carrier frequencies, and perform coherent demodulation and/or analysis of each. In this section, we assume that there is only one instantaneous carrier frequency, whose modulation spectrum can be derived using fast non-coherent demodulation.

Non-coherent demodulation begins with envelope detection, $|x[n]|$, where $x[n]$ is the audio signal, and $||$ denotes absolute value. A standard envelope detector extracts the envelope as a signal in its own right, by immediately lowpass filtering $|x[n]|$. The goal of auditory roughness detection is not, however, to perform an exhaustive analysis of the spectrum of $|x[n]|$; rather, we simply want to identify components of that spectrum in the neighborhood

of 75Hz. To do so, we modulate several different frequency components down to baseband:

$$e_k[n] = |x[n]| \sin(\Omega_k n) \tag{2.1}$$

where Ω_K are a set of K different modulation frequencies in the neighborhood around 75Hz. Instantaneous roughness, $y[n]$, is then defined to be the weighted sum of demodulated envelopes:

$$y[n] = \sum_{k=1}^{K/2} w_k |x[n]| \left(\sin[\Omega_k n] + \sin[\Omega_{K+1-k} n]\right) \tag{2.2}$$

In Eq 2.2, Ω_k are frequencies near $75 * 2\pi/Fs$, chosen symmetrically so that $\Omega_{K+1-k} = 300\pi/Fs - \Omega_k$ and with symmetric combination weights $w_{K+1-k} = w_k$. To smooth out the signal, $y[n]$ is lowpass filtered to create the smoothed roughness signal $z[n]$:

$$z[n] = \sum_{m=0}^{255} b[m] y[n-m] \tag{2.3}$$

where $b[m]$ are the coefficients of a 256-tap FIR filter with a cutoff of 62.5Hz, and F_s is the sampling frequency. The smoothed roughness signal $z[n]$ is then downsampled by a factor of 128, to a sampling frequency of 62.5Hz. In order to match the output range of the original algorithms, its absolute value is computed as the approximated auditory roughness $A[n]$:

$$A[n] = |z[n]| \tag{2.4}$$

In our approximation, Eq 2.4 requires either one or no additions, and no multiplications. The FIR filter in Eq 2.3 requires 256 real MAC. The weighted summation in Eq 2.2 requires $K/2 = 2$ real MAC per sample for $K = 4$, which returns very similar results compared to larger K value. Since all factors in Eq 2.2 operate on the $|x[n]|$, we could pre-compute the result and store $\sum_{k=1}^{K/2} w_k \left(\sin[\Omega_k n] + \sin[\Omega_{K+1-k} n]\right)$ in our look-up-table, reducing the complexity of this operation down to 1MAC/sample. When we implement Eq 2.3 using a

multi-phase filter, the downsampling operations allow us to carry out both Eq 2.2 and 2.3 once every 128 samples, thus requiring only $256 * (1 + 1)/128 \approx 4$MAC/sample. If the hardware specification allows, we can duplicate the filter coefficients for all possible multiplets in Eq 2.2; this brings the complexity down to 2MAC/sample, but at the cost of more memory. In practice, since our measure is not at all sensitive to phase, we can reduce memory usage at the cost of phase discontinuity. The Goertzel algorithm [40] is a promising alternative for extracting the power around the frequency of interest. The Goertzel algorithm requires N real MAC and 1 CMAC to extract the power of a single discrete frequency, where N is the DCT window size. When we are only interested in 1 frequency component, the Goertzel algorithm has an advantage. But this advantage vanishes when the components of interest exceed 2. The final computation has a complexity level similar to STE, which requires 1 MAC per sample. In comparison, mel-frequency cepstral coefficients (MFCCs) require the computation of a full FFT once per frame; a 256-sample FFT computed once per 128 samples requires $256 \log_2(256)/128 = 16$ complex multiply-accumulate (CMAC) operations per sample, not including the filterbank accumulation, DCT, delta and second delta extractions. If we consider the remaining operation, the total complexity will not be less than 20 CMAC; the proposed approximated roughness calculation is at least $10\times$ less complex than MFCC, and is therefore well-suited to a sensor hub with no dedicated multiplier and with dozens of sensors to monitor.

2.3 Results

2.3.1 Performance of the AED

As introduced before, we implemented several promising classifiers. The first classifier is a decision tree, which evaluates the DSP features against a set of thresholds determined at training time. We also implemented and tested an artificial neural network (ANN). The ANN consists of 3 fully connected layers of 16 neurons, 4 neurons and 1 neuron, respectively.

To fit it into the FPGA, we implemented the ANN around a single fixed-point multiplier unit of 16-bit, using a state machine to time-share the multiplier unit. An accuracy comparison between the ANN and decision tree is shown in Fig 2.7, which plots the true positive (TP) rate as a function of the false positive (FP) rate. Note this experiment used sound samples with noisy background. When background noise level is low, the FP rate can be considerably lower. We also implemented a LSTM classifier, which demonstrated 2% better accuracy over ANN. However, it used a large amount of resource so we cannot use it for the current FPGA. For the sound localization module, the FPGA constantly keeps a circular buffer of stereo audio samples. When a suspicious event is determined to have happened, the localization module computes the minimum delayed difference between the left and the right channels, yielding the number of samples by which the left channel is delayed from the right, which may be positive or negative. This is different from the usual cross-correlation method, which requires the use of a fast multiply-accumulate (MAC) module leading to a high resource usage. Our design is much leaner but yields similar results. After a positive event is detected, the FPGA uses the XBee X2 wireless module to transmit the direction and localization data to a host computer.

2.3.2 Mandarin Affective Speech

The Mandarin Affective Speech corpus [33] includes short phrases and sentences recorded from 68 speakers under different emotional states. Three of these emotions are neutral, anger and panic. One phrase is repeated by the same person under all emotion states. All audio files are normalized to have the same maximum amplitude.

In our experimental setup, we extracted a collection of features from every phrase under all emotional states. Features from every phrase pronounced under anger and panic were compared to the same phrase spoken neutrally. The five features include auditory roughness, which we extracted using open access Matlab toolbox MIRtoolbox [38], approximated roughness, STE, and the signal remainder after subtraction of its windowed media (SWM, [41]).

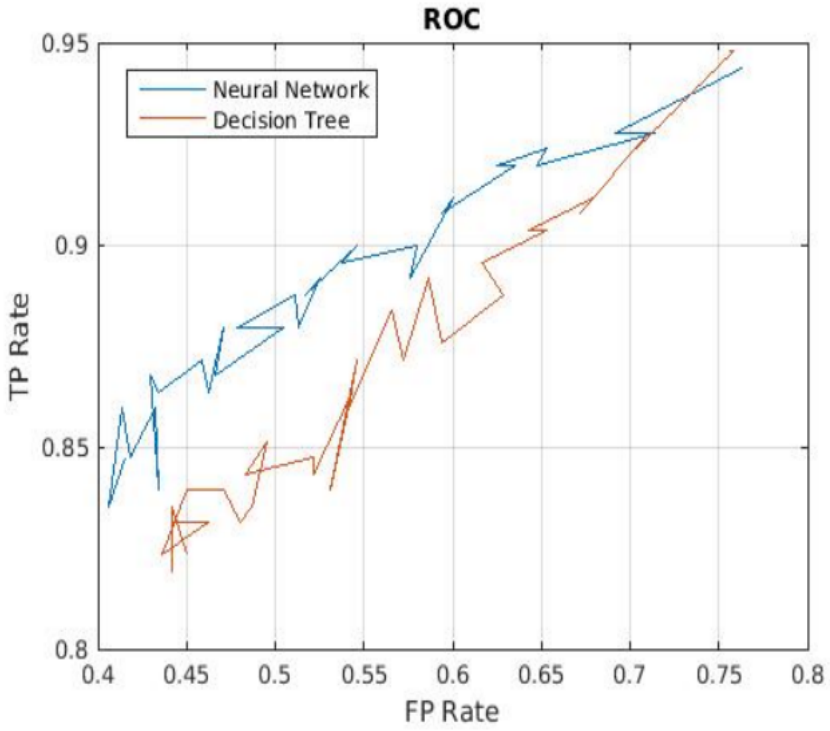


Figure 2.7: Classifier accuracy comparison for the AED.

Zero crossing rate (ZCR) was also included as it is a common feature for speech detection [42] and used in many previous papers to detect screaming and/or emotional speech [24, 6, 25, 26]. Figure 2.8 presents an example of the features for a short phrase spoken by the same person under three different emotional status. As we can see from Fig 2.8, STE and SWM fail to effectively separate affective speech, especially angry speech, from neutral. STE is closely related to the loudness of the original audio, and as a result, natural speech recorded at close distance will have high STE as well. SWM worked mainly through detecting impulse in the audio waveform; however, affective speech shows limited difference in this measure. On the other hand, the auditory roughness value of angry speech is much larger than that of neutral speech. The maximum value of the angry speech is many times higher than the neutral speech. One can observe the shape of the auditory roughness is different from any other measure: it does not resemble the envelope of the audio input. Our approximated roughness cannot reach the same level of separation as the original auditory roughness, yet

it still records observably higher peaks for angry and frightened speech. One could also observe, throughout the audio, auditory roughness and our approximated roughness do not maintain a high value, but rather peak out in the middle.

We ran through all short phrases in the corpus. Figure 2.9 presents overall statistics and confirms our finding above. The left-most boxplot, for example, represents the ratio between 2 maximum values, one of which is for the feature extracted from an angry speech, the other being features extracted from the same speech under neutral state.

The 5 letters in Fig 2.9 represent auditory roughness (R), approximated auditory roughness (A), STE (E), SWM (M) and ZCR (Z). In all cases, a significant difference [43], with confidence level 0.005, can be found between the max value of roughness in any given audio file (both Auditory Roughness and our approximated roughness) in angry versus neutral speech. However, we can see roughness is less effective in separating panic and neutral speech. Also the average value of roughness, in any given waveform file, is less effective in separating affective versus neutral speech.

2.3.3 Youtube AudioSet

To test our approximated roughness in a more realistic setup, we built a simple test on the Google AudioSet [34] for screaming. This dataset contains video recordings of screaming and shouting, many happening in real life and recorded through smart phones. The events are typically shorter than scream events in movies and other sources, but the recording quality varies from video to video. Annotation is at a relatively coarse temporal granularity: a window of 10s is marked in every video, which either contains some short affective speech events, separated by pauses, or contains non-stop screaming extending outside the 10s window. We found this difficult to use as the event boundaries are not comprehensive, and markings for non-screaming regions are not presented. We took some time to conduct more fine-grained annotation on the balanced training subset and extended the duration of the feature window from 10s to 30s to include non-screaming regions. A couple of the files were dropped as they

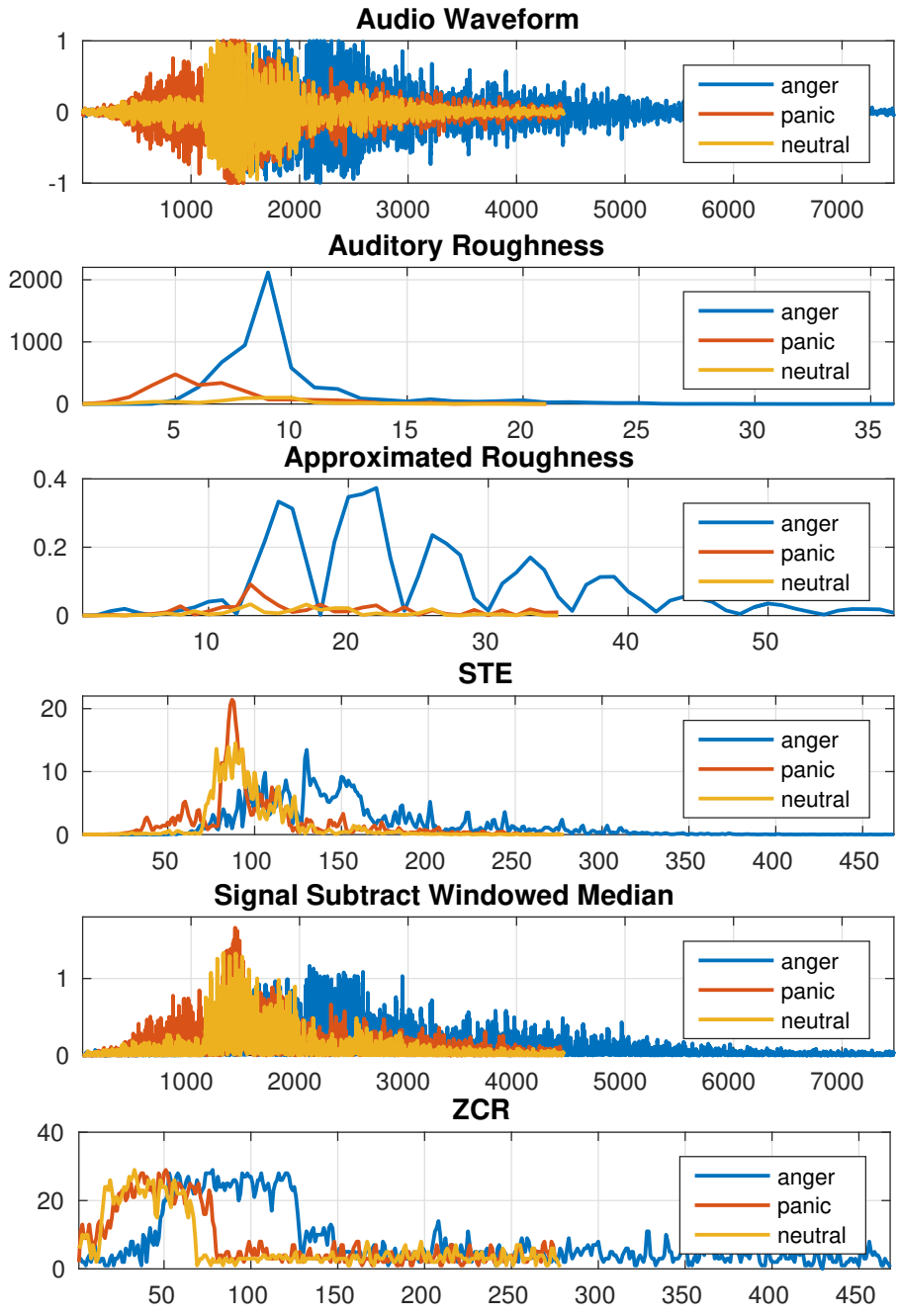


Figure 2.8: Feature measures for the same short phrase by the same person under different emotional states.

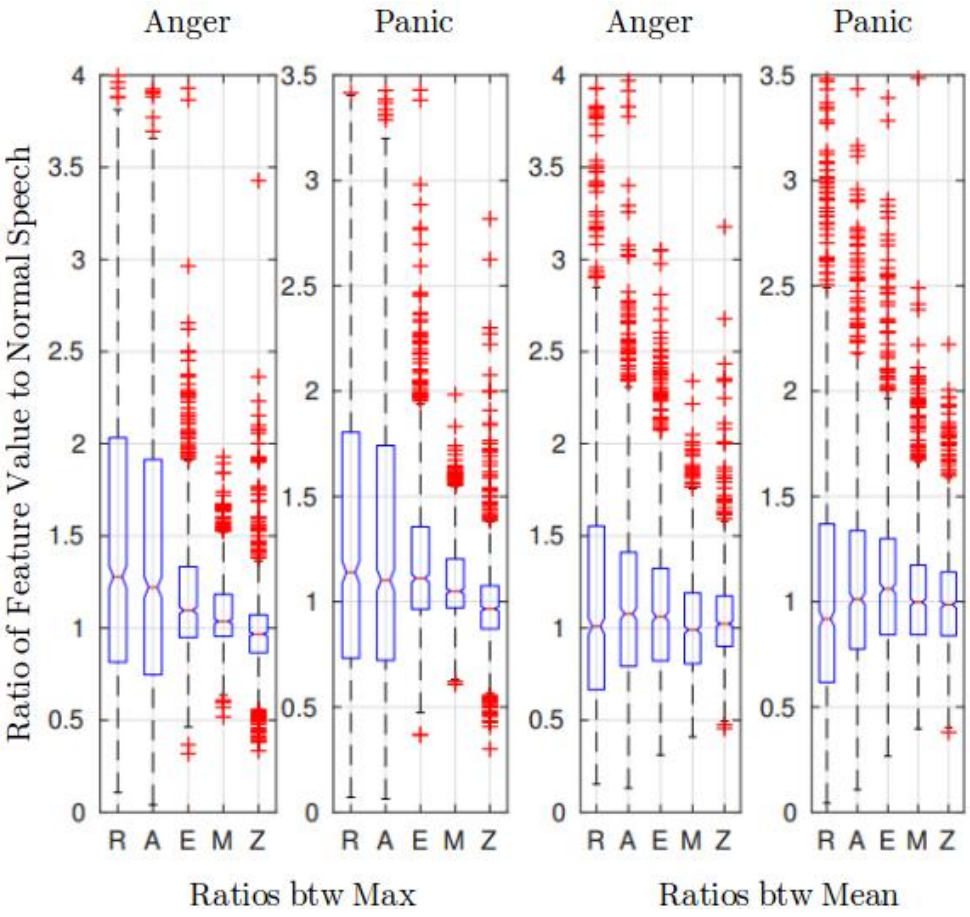


Figure 2.9: Ratio between maximum (left 2) mean (right 2) values of angry to neutral (1, 3) and panic to neutral speech.

clearly contain no human vocalizations. We annotated 55 files with a total of 0.5h duration. Since our features are extracted at relatively high rate, this transfers to $112k$ sample. About 33% of the 0.5h audio is screaming or affective speech; this is a larger dataset than any other open-access corpus of human scream audio we found. These annotations are available online, [2] and will be released with a creative commons CC-BY license.

We designed a simple experiment to mimic the pre-filter operation explained in Sec 2.1. We marked out screaming as audio events of interests and labeled the begin and end of each event. For each event in the video, if the feature value exceeds a pre-selected threshold, we considered the event detected; during the time between events, feature values exceeding the threshold are considered false alarms. Any event longer than 2s is separated into multiple events. The ROC curves comparing different features are presented in Fig 2.10. We sweep through different threshold value to obtain this ROC curve. It is worth mentioning, in most videos, human vocal do not span the full duration, yet we did not run speech activity detection on top of the algorithm. Doing so would add more computation and therefore undermine the purpose of pre-filtering. As we can see in Tab 2.2, this made the detection task difficult even for relatively complex feature and classifiers.

We can see from Fig 2.10 that, mostly, our approximated roughness returns better value than all other features. To our surprise, our approximated roughness actually outperforms the original auditory roughness; the original auditory roughness measure has the advantage only in limited cases. We suspect this is because the majority of the events in our collection are screaming and shouting, making them more fitted to the category described in [35], and less similar to affective speech. The equal error rate for our approximated roughness is around 30%.

[2] `https://github.com/dihe2/Audioset-Balanced-Training-Annotation`

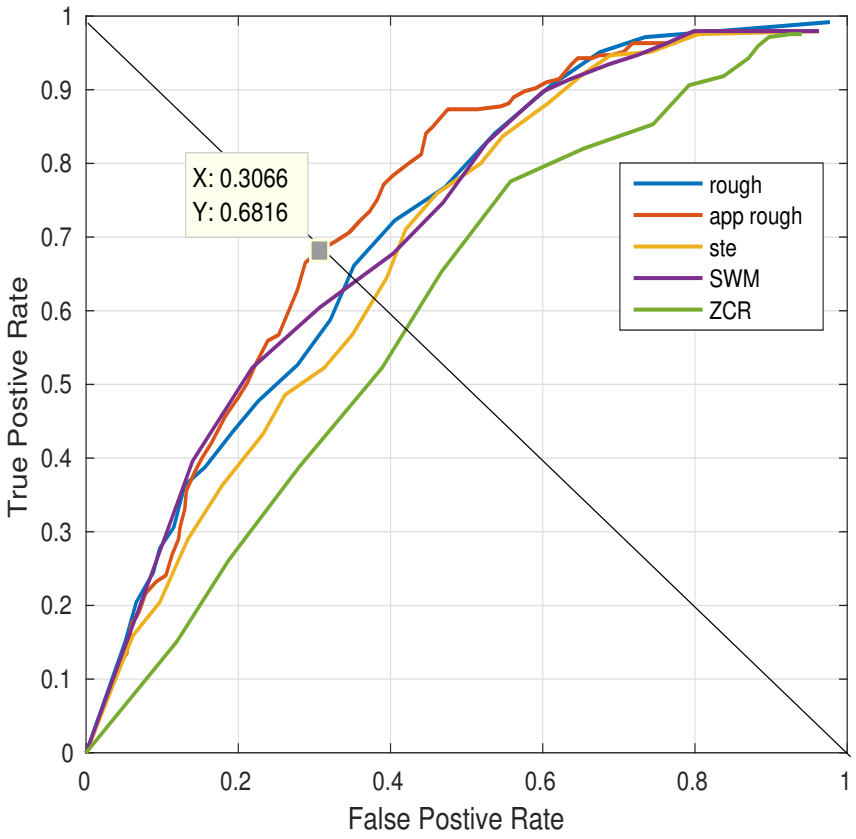


Figure 2.10: ROC curves of testing different features on AudioSet.

2.3.4 Beyond Pre-filtering

We ran more experiments, in the interest of answering the following two more questions. First, how hard is our AudioSet corpus? The dataset was released very recently, and not much work has been published using it. Second, how well will our low complexity feature work in a classifier, say linear SVM; and how will it compare to high complexity features like MFCC? We stacked 5 frames, each containing approximated roughness, STE and ZCR and used a linear SVM to conduct a detection task. Our low complexity three-dimensional feature vector is compared to a standard MFCC vector extracted at 50ms per frame. As we can see in Tab 2.2, MFCC returns reasonable results compared to previous work, though with accuracy below that of most previous affective speech studies, suggesting that this corpus is difficult to classify. A feature vector with only STE and ZCR achieves an F1 score about 10% worse than that of MFCC; the same feature vector with approximated auditory roughness included is only 8% worse than MFCC. The MFCC are extracted using the open source toolbox RASTA for Matlab [37] and the SVM is trained using SVMLight [44]. More could be compared between MFCC and our low-complexity features. But as the goal of this work is to study the potential of our approximated roughness, more detailed experiment on a complex classifier is beyond this work.

Table 2.2: Linear SVM on AudioSet.

	stack5(wo Rough)	stack5(w Rough)	MFCC
Prec	65.12%	67.25%	72.13%
F1	63.34%	65.47%	73.73%

2.4 Recapitulation

In this work, we evaluated the auditory roughness as a feature for pre-filtering the audio for AED targeting human screaming and affective speech. Detecting these events is of interest to distributed security and surveillance AED systems. In order to be useful in a large distributed

system, detection must have extremely low computational cost; MFCC is too expensive. We proposed a method to approximate roughness using a combination of frequency modulation and multiphase filtering. This allows us to extract a feature with computational cost similar to STE. We proved through experiments on the Mandarin Affective corpus, and a subset of the Google AudioSet, that our approximated roughness also is more accurate than other low-complexity features.

Chapter 3

Using Acoustic Landmarks to Improve ASR through Frame Dropping and Frame Re-weighting

Ideas from speech science – which may have the potential to further improve modern automatic speech recognition (ASR) – are not often applied to ASR systems [1]. Speech science has demonstrated that perceptual sensitivity to acoustic events is not uniform in either time or frequency. Most modern ASR uses a non-uniform frequency scale based on perceptual models such as critical band theory [45]. In the time domain, however, most ASR systems use a uniform or *frame synchronous* time scale: systems extract and analyze feature vectors at regular time intervals, thereby implementing a model according to which the content of every frame is equally important.

Acoustic landmark theory [18, 19] is a model of experimental results from speech science. It exploits quantal nonlinearities in articulatory-acoustic and acoustic-perceptual relations to define instances in time (landmarks) at which abrupt changes or local extrema occur in speech articulation, in the speech spectrum, or in a speech perceptual response. Landmark theory proposes that humans perceive phonemes in response to acoustic cues, and that such cues are anchored temporally at landmarks, i.e., that a spectrotemporal pattern is perceived as the cue for a distinctive feature only if it occurs with a particular timing relative to a particular type of landmark. Altering distinctive features alters the phone string; distinctive features in turn get signaled by different sets of cues anchored at landmarks.

The theory of acoustic landmarks has inspired a large number of ASR systems. Acoustic landmarks have been modeled explicitly in ASR systems such as those reported by [46, 47, 48]. Many of these systems have accuracies comparable to other contemporaneous systems - in some cases, even returning better performance [46]. However, published landmark-based

ASR with accuracy comparable to the state of the art has higher computation than the state of the art; conversely, landmark-based systems with lower computational complexity tend to also have accuracy lower than the state of the art. No implementation of acoustic landmarks has yet been demonstrated to achieve accuracy equal to the state of the art at significantly reduced computational complexity. If acoustic landmarks contain more information about the phone string than other frames, however, then it should be possible to significantly reduce computational complexity of a state of the art ASR without significantly reducing accuracy, or conversely, to increase accuracy without increasing computation, by forcing the ASR to extract more information from frames containing landmarks than from other frames.

We assume that a well trained frame-synchronous statistical acoustic model (AM), having been trained to represent the association between MFCC features and triphones, has also learned sufficient cues and necessary contexts to associate MFCCs and distinctive features. However, because the AM is frame-synchronous, it must integrate information from both informative and uninformative frames, even if the uninformative frames provide no gain in accuracy. The experiments described in this chapter explore whether, if we treat frames containing acoustic landmarks as more important than other frames, we can get better accuracy or lower computation. In this work, we present two methods to quantify the information content of acoustic landmarks in an ASR feature string. In both cases, we use human annotated phone boundaries to label the location of landmarks. The first method seeks to improve ASR accuracy by over-weighting the AM likelihood scores of frames containing phonetic landmarks. By over-weight, we mean multiplying log-likelihoods with a value larger than 1 (Sec 3.2.1). The second method seeks to reduce computation, without sacrificing accuracy, by removing frames from the ASR input. Removing frames makes the computational load decrease, but usually causes accuracy to decrease also: Which frames can be removed that cause the accuracy to drop the least? We searched for a strategy that removes as many frames as possible while attempting to keep the phone error rate (PER) low. We show that if we know the locations of acoustic landmarks, and if we retain these

frames while dropping others, it is possible to reduce computation for ASR systems with a very small error increment penalty. This method for testing the information content of acoustic landmarks is based on past works [49, 50, 51] that demonstrated significantly reduced computation by dropping acoustic frames, with small increases in PER depending on the strategy used to drop frames. In this chapter we adopt the PER increment as an indirect measure of the phonetic information content of the dropped frames.

If the computational complexity of ASR can be reduced without sacrificing accuracy, or if the accuracy can be increased without increasing the computational load, these findings should have practical applications. It is worth emphasizing that this work only intends to explore these potential applications, assuming landmarks can be accurately detected. Our actual acoustic landmark detection accuracy, despite increasing over time, has not reached a practical level yet.

Section 3.1 briefly reviews the acoustic landmark theory and relevant works which apply it to ASR systems. Sec 3.2 presents the theoretical basis for our experiments. Section 3.3 proposes the hypothesis. Experimental setup is explained in Sec 3.4 and results are presented in Sec 3.5. Discussion, including a case study of the confusion characteristics, is presented in Sec 3.6. At last, our conclusions are drawn in Sec 3.7.

3.1 Background and Literature Review

Acoustic landmark theory was first proposed as a theory of the perception of distinctive features; therefore many landmark-based ASRs use distinctive features rather than triphones [52] as their finest-grain categorical representation. Distinctive features are an approximately binary encoding of perceptual [53], phonological [54], and articulatory [18] speech sound categories. A feature is called "distinctive" if and only if it defines a phoneme category boundary; therefore distinctive features are language dependent. The distinctive features used by each language often have articulatory, acoustic, and/or perceptual correlates

that are similar to those of distinctive features in other languages, however [55, 56], so it is possible to define a set of approximately language-independent distinctive features as follows: if an acoustic or articulatory feature is used to distinguish phonemes in at least one of the languages of the world, then that feature may be considered to define a language-independent distinctive feature. Each phoneme of a language is a unique vector of language-dependent distinctive features. Automatic speech recognition may distinguish two different allophones of the same phoneme as distinct phones; in most cases, the distinctions among phones can be coded using distinctive features borrowed from another language, or equivalently, from the language-independent set.

The ASR community has explored a number of encodings similar to distinctive features, e.g., articulatory features [57, 58, 59, 60, 61, 62] and speech attributes [63]. These concepts have different foci, but are also very similar. Distinctive features are defined by phoneme distinctions; therefore they are language dependent. It is possible to define a language-independent set of distinctive features based on quantal nonlinearities in the articulatory-acoustic [64] and acoustic-perceptual [18] transformations. Although both the articulatory-acoustic and acoustic-perceptual transformations contain quantal nonlinearities that may define distinctive features, a much larger number of nonlinearities in the articulatory-acoustic transformation has been demonstrated. Many studies therefore focus only on the set of phoneme distinctions defined by nonlinearities in the articulatory-acoustic transformation, which are called "articulatory features" in order to denote their defining principle. Of these, some studies focus on the articulatory-acoustic transform because it implies a degree of acoustic noise robustness [57, 58, 59], others because it implies a compact representation of pronunciation variability [60], others because it is demonstrably language-independent [61, 62]. Speech attributes, on the other hand, are a super-set of distinctive features; they are deliberately defined to introduce other purposes to speech recognition. In Lee's framework [63], speech attributes are quite broadly defined to be perceptible speech categories, of which phonological categories are only a subset. Under this definition, speech

attributes include not only distinctive feature but also a wide variety of acoustic cues signaling gender, accent, emotional state and other prosodic, meta-linguistic, and para-linguistic messages.

As opposed to modern statistical ASR where each frame is treated with equal importance, landmark theory proposes that there exist information rich regions in the speech utterance, and that we should focus on these regions more carefully. These regions of interest are anchored at acoustic landmarks. Landmarks are instantaneous speech events near which distinctive features are most clearly signaled. These key points mark human perceptual foci and key articulatory events [65]. Stevens [18] first introduced these instantaneous speech events, where, for some phonetic contrasts, humans focus their attention in order to extract acoustic cues necessary for identifying the underlying distinctive features. Initially Stevens named these key points "acoustic boundaries"; the name "acoustic landmarks" was introduced in 1992 (Stevens1992), and has been used consistently since. At roughly the same time [66] and [67] made similar observations when studying children's speech perception in Japanese.

Liu [65] demonstrated algorithms for automatically detecting acoustic landmarks. Hasegawa-Johnson [68] measured the phonetic information content of known acoustic landmarks. He defined a set of landmarks including consonant releases and closures (at phone boundaries) and vowel/glide pivot landmarks (near the center of the corresponding phones). In contrast, Lulich [69] argued that the center of vowels and glides are not as informative and should not be considered as landmarks. He defined, instead, formant-subglottal resonance crossing, which is known to sit between boundaries of [-back] and [+back] vowels, to be more informative. Wang et al. [70] showed that the latter improves performance for automatic speaker normalization application. Hasegawa-Johnson [68] defined a small number of pivot and release landmarks at +33% and −20% locations after the beginning or before the end of certain phones (where +33% indicates delaying the location by 33% of the total duration of that phone; −20% indicates advancing the location by 20%), in order to better approximate

the typical timing of the spectrotemporal events defined earlier in Liu's work [65]. Later works [46, 71] labeled these landmarks right on the boundary and returned performance similar to that of [68]. Figure 3.1 illustrates the landmark labels for the pronunciation of word "Symposium".[1] The details of the landmark labeling heuristics applied in this example are further described in Tab 3.1.

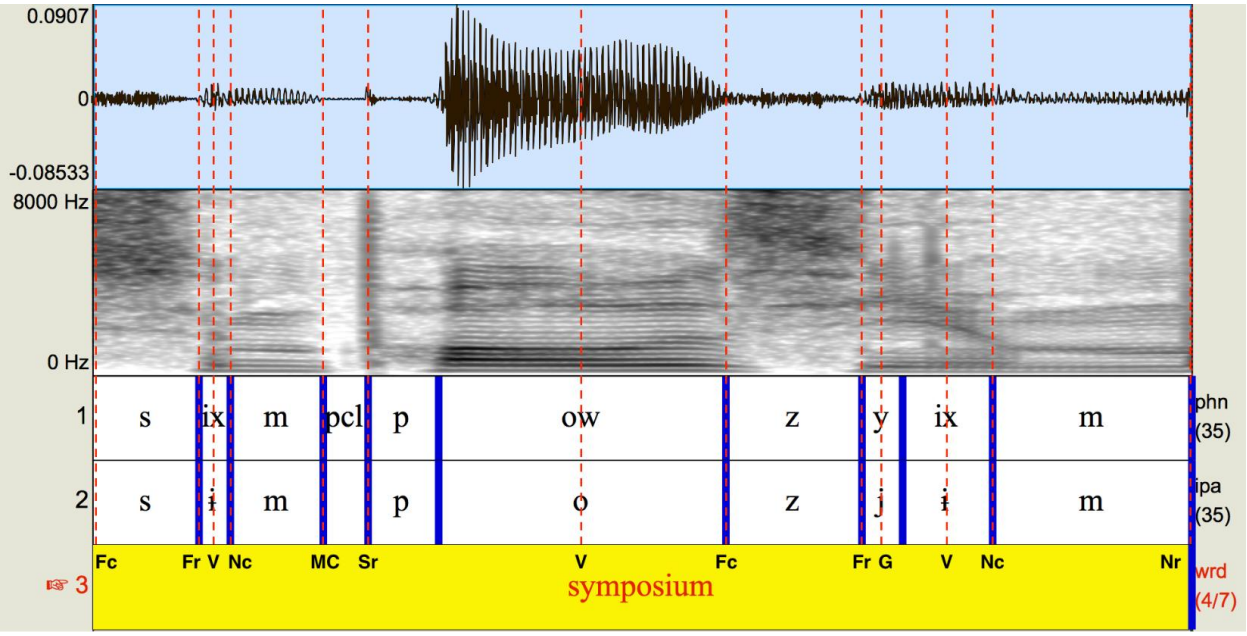


Figure 3.1: Acoustic landmark labels (LM) for the pronunciation of the word "Symposium". TIMIT phone symbols (PHN) and international phonetic alphabet (IPA) symbols are both used in this example. The dashed red lines denote the landmark positions. The symbols under the dashed red lines are landmark types, where **Fc** and **Fr** are closure and release for fricatives; **Sc** and **Sr** are closure and release for Stops; **Nc** and **Nr** are closure and release for nasals; **V** and **G** are vowel pivot and glide pivot; **MC** is manner-change landmark.

Many works have focused on accurately detecting acoustic landmarks. The first of these assumed that landmarks correspond to the temporal extrema of energy or energy change in particular frequency bands. For example, Liu [65] detected consonantal landmarks in this way, [72] detected vowel landmarks, [73] classified consonant voicing, and [74, 75, 76, 77] classified place of articulation. Support vector machines (SVMs) were popularized for landmark detection by [78], who showed that an SVM trained to observe a very small

[1]The pronunciation of "Symposium" is selected from audio file: `TIMIT/TRAIN/DR1/FSMA0/SX361.WAV`

acoustic feature vector (only four measurements, computed once per millisecond) can detect stop release landmarks more accurately than a hidden Markov model. Both [79] and [80] target the detection of all landmarks using one kind of acoustic features. Their results are reasonably accurate, but are still less accurate and more computationally expensive than the best available classifier for each distinctive feature. Xie and Niyogi [81] expanded the work of [78] by demonstrating detection of several different types of landmark using a very small acoustic feature vector. In Qian's paper [82], a small vector of acoustic features was learned, using the technique of local binary patterns, and resulting in accuracy above 95% for stop consonant detection. In a paper from [71], a convolutional neural network (CNN) trained on MFCC and additional acoustic features achieved around 85% on consonant voicing detection. This system was trained on the English corpus TIMIT [83], but tested on Spanish and Turkish corpora. Over time, new techniques and more specific features have been developed for landmark detection, and the detection accuracy has been improving steadily. Acoustic landmarks were first introduced as part of an ASR in 1992 [84], and have been used in a variety of ASR system architectures. These systems, without considering the mechanism used for landmark detection, can be clustered into two types. The first type of system, such as those described by [65, 48, 47], computes a lexical transcription directly from a set of detected distinctive features. Due to the complexity of building a full decoding mechanism on distinctive features, some of these systems only output isolated words. However, other systems (e.g.,work from [48]) have full HMM back-ends that can output word sequences. The other type of system, such as that described by [46], conducts landmark-based re-scoring on the lattices generated by an MFCC-based hidden Markov model. Acoustic likelihoods from the classic ASR systems are adjusted by the output of the distinctive feature classifier. Many landmark based ASRs demonstrated performance slightly [46] or even significantly [57] better than baseline ASR systems, especially in noisy conditions.

3.2 Measures of the Information Content of Acoustic Frames

An acoustic landmark is an instantaneous event that serves as a reference time point for the measurement of spectrotemporal cues widely separated in time and frequency. For example, in the paper that first defined landmarks, Stevens proposed classifying distinctive features of the landmark based on the onsets and offsets of formants and other spectrotemporal cues up to 50ms before or 150ms after the landmark [18]. The 200ms spectrotemporal dynamic context proposed by Stevens is comparable to the 165ms spectrotemporal dynamic context computed for every frame by the ASR system of [85]. Most ASR systems use acoustic features that are derived from frames 25ms long, with a 10ms skip, as human speech is quasi-stationary for this short period [86]. Because spectral dynamics communicate distinctive features, however, ASR systems since 1981 [87] have used dynamic features; since deep neural nets (DNNs) began gaining popularity, the complexity of the dynamic feature set in each frame has increased quite a lot, with consequent improvements in ASR accuracy. This trend not only applies to stacking below 100ms. With careful normalization, features like TRAPs [88], with temporal window equal or longer than 500ms, continue to demonstrate accuracy improvement. Experiments reported in this chapter are built on a baseline described by [85], and schematized in Fig 3.2. In this system, MFCCs are computed once every 10ms, with 25ms windows (dark gray rectangles in Fig 3.2). In order to include more temporal context, we stack adjacent frames, three preceding and three succeeding, for a total of seven frames (a total temporal spanof $(7-1)\times 10+25=85$ ms). These are shown in Fig 3.2 as the longer, segmented dark gray rectangles, with the red segments representing the center frames of each stack. The seven-frame stack is projected down to 40 dimensions using linear discriminant analysis (LDA). For input to the DNN but not the GMM, LDA is followed by speaker adaptation using mean subtraction and feature-space maximum likelihood linear regression, additional context is provided by a second stacking

operation afterwards, in which LDA-transformed features, represented by yellow rectangles, are included in stacks of 9 frames (for a total temporal span of $(9-1)\times 10+85=165$ms), as represented by the top path in Fig 3.2. It is believed that the reason features spanning longer duration improve ASR accuracy is that long lasting features capture coarticulation better, including both neighboring-phone transitions and longer-term coarticulation. The dynamics of the tongue naturally cause the articulation of one phoneme to be reflected in the transitions into and out of neighboring phonemes, over a time span of perhaps 70ms. Longer-term coarticulation, spanning one or more syllables, can occur when an intervening phoneme does not require any particular placement of one or more articulators; e.g., [89] demonstrated that the tongue body may transition smoothly from one vowel to the next without apparently being constrained by the presence of several intervening consonants.

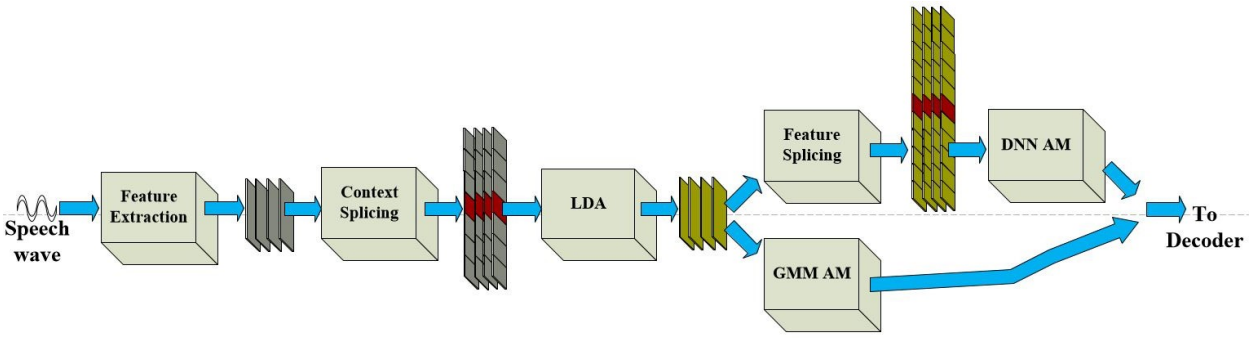


Figure 3.2: Stacking of feature frames before the scoring process for DNN AM (top path) and GMM AM (bottom path). The dark gray, red and green rectangles indicate frames and stacks of frames.

3.2.1 Frame Re-weighting

HMM-based ASR searches the space of all possible state sequences for the most likely state sequence given the observations. During the state likelihood estimation, results of all frames are weighted equally. Weighting more informative frames more heavily could potentially benefit speech recognition. Ignoring the effects of the language model, the log-likelihood of

a state sequence S given the observations O is

$$L(S|O) = \sum_{t=1}^{T} w(t) log(p(o_t|s_t)) + log(p(s_t|s_{t-1})) \tag{3.1}$$

where s_t and o_t are respectively the state and observed feature vector associated with the frame at time t. The state s_t at any time should be associated with one of the senones (i.e., monophone or clustered triphone states). Here $p(s_t|s_{t-1})$ is the transition probability between senones, which we will not consider modifying in this study. In most systems, beam search parameters constrain the number of active states, thus we only need to evaluate the necessary posteriors. In our over-weighting framework, if o_t contains a landmark, the value of $\log p(o_t|s_t)$ will be scaled. To simplify the computation, we operate directly on log-likelihoods. In this case, $log(p(o_t|s_t))$ is multiplied by factor $w(t)$ which takes the value 1 when frame t contains no landmark and a value greater than 1 otherwise. This is effectively applying a power operation on the likelihoods.

The key in this strategy is that the likelihood of all model states will be re-weighted. If the frame over-weighted is a frame that can differentiate the correct state better, the error rate will drop. In contrast, if the likelihood of a frame is divided evenly across states, or even worse, is higher for the incorrect state, then over-weighting this frame will mislead the decoder and increase chances of error. For this reason, over-weighting landmark frames is a good measure to tell how meaningful landmark frames are compared to the rest of the frames. If the landmarks are indeed more significant, we should observe a reduction in the PER for the system over-weighting the landmark.

3.2.2 Frame Dropping

The wide temporal windows used in modern ASR, as mentioned in the beginning of Sec 3.2, are highly useful to landmark-based speech recognition: all of the dynamic spectral cues proposed by [18] are within the temporal window spanned by the feature vector of a frame

centered at the landmark; therefore it may be possible to correctly identify the distinctive features of the landmark by dropping all other frames, and keeping only the frame centered at the landmark. Our different frame dropping heuristics modify the log probability of a state sequence by replacing the likelihood $p(o_t|s_t)$ with an approximation function f. In terms of log probabilities, Eq (3.1) becomes

$$L(S|O) = \sum_{t=1}^{T} logf(p(o_t|s_t), t) + log(p(s_t, s_{t-1})) \tag{3.2}$$

The class of optimizations considered in this chapter involve a set of functions $f(p(o_t|s_t))$ parameterized as:

$$f(p(o_t|s_t)) = \begin{cases} R(O,t) & \text{if } g(t) = 1 \\ p(o_t|s_t) & \text{otherwise} \end{cases} \tag{3.3}$$

The *method of replacement* is characterized by R, and the frame-dropping function by $g(t)$. This work considers multiple methods to verify that the finding with respect to landmarks is independent of the replacement method. The four possible settings of the $R(o,t)$ function are as follows:

$$R(O,t) \in \begin{cases} R_{\text{Copy}}(O,t) & = p(o_{t'}|s_{t'}), \quad t' = \max_{\tau \le t,\ g(\tau)=0} \tau \\ R_{\text{Fill\_0}}(O,t) & = 1 \\ R_{\text{Fill\_const}}(O,t) & = \left(\prod_{t=1}^{T} p(o_t|s_t)\right)^{1/T} \\ R_{\text{Upsample}}(O,t) & = \exp\left(\sum_{t':g(t')=0} h(t-t') \log p(o_t|s_t)\right) \end{cases} \tag{3.4}$$

In other words, the `Copy` strategy copies the most recent observed value of $p(o_t|s_t)$, the `Fill_0` strategy replaces the log probability by 0, the `Fill_const` strategy replaces the log probability by its mean value, and the `Upsample` strategy replaces it by an interpolated value computed by interpolating (using interpolation filter $h(t)$) the log probabilities that have been selected for retention. The `Upsample` strategy will only be used if the frame-

dropping function is periodic, i.e., if frames are downsampled by a uniform downsampling rate.

The *pattern of dropped frames* can be captured by the indicator function g, which is true for frames that we want to drop. Experiments will test two landmark-based patterns: `Landmark-drop` drops all landmark frames ($g(t) = 1$ if the frame contains a landmark), and `Landmark-keep` keeps all landmark frames ($g(t) = 1$ only if the frame does *not* contain a landmark). In the case where landmark information is not available, the frame-dropping pattern may be `Regular`, in which $g(t) = \delta(\ t \bmod K\ )$ indicates that every K-th frame is to be dropped, or it may be `Random`, in which case the indicator function is effectively a binary random variable set at a desired frame dropping rate. As we will demonstrate later, to achieve a specific function and dropping ratio, we can sometimes combine output of different g functions together by taking a logical inclusive OR to their output.

If acoustic landmark frames contain more valuable information than other frames, it can be expected that experiment setups that retain the landmark frames should out-perform other patterns, while those that drop the landmark frames should under-perform, regardless of the *method of replacement* chosen.

3.3 Hypotheses

This chapter tests two hypotheses. The first is that a window of speech frames (in this case 9 frames) centered at a phonetic landmark has more information than windows centered elsewhere – this implies that over-weighting the landmark-centered windows can result in a reduction in PER. The second hypothesis states that keeping landmark-centered windows rather than other windows causes little PER increment, and that dropping a landmark-centered window causes greater PER increment as opposed to dropping other frames. In the study we focused on PER as opposed to word error rate (WER) for two reasons. First, the baseline Kaldi recipe for TIMIT reports PER. Second, this study is oriented towards

speech acoustics; focusing on phones allowed us to categorize and discuss the experiment and results in better context.

Table 3.1: Landmark types and their positions for acoustic segments. **Fc** and **Fr** are closure and release for fricatives; **Sc** and **Sr** are closure and release for Stops; **Nc** and **Nr** are closure and release for nasals; **V** and **G** are vowel pivot and glide pivot; 'start', 'middle', and 'end' denote three positions across acoustic segments.

Manner of Articulation	Landmark Type and Position	Observation in Spectrogram
Vowel	V: middle	maximum in low- and mid-frequency amplitude
Glide	G: middle	minimum in low- and mid-frequency amplitude
Fricative	Fc: start, Fr: end	amplitude discontinuity occurs when consonantal constriction is formed or released
Affricate	Sr,Fc: start, Fr: end	
Nasal	Nc: start, Nr: end	
Stop	Sc: start, Sr: end	

In order to test these hypotheses, a phone boundary list from the TIMIT speech corpus [83] was obtained, and the landmarks were labeled based on the phone boundary information. Table 3.1 briefly illustrates the types of landmarks and their positions, as defined by the TIMIT phone segments. This marking procedure is shared by [56, 46, 71]. It is worth mentioning that this definition disagrees with that of [69]. Lulich claims that there is no landmark in the center of vowel and glide; instead, a formant-subglottal resonance crossing, which is known to sit between the boundaries of [-Back] and [+Back] vowels, contains a landmark. Frames marked as landmark are of interest. To test hypothesis 1, landmark frames are over-weighted. To test hypothesis 2, either non-landmark or landmark frames are dropped.

3.4 Experimental Methods

Our experiments are performed on the TIMIT corpus. Baseline systems use standard examples distributed with the Kaldi open source ASR toolkit.[2] Specifically, the GMM-based baseline follows the configurations in the distributed `tri2` configuration in the Kaldi TIMIT

[2] `http://kaldi-asr.org/`

example files.[3] The clustered triphone models are trained using maximum likelihood estimation of features that have been transformed using linear discriminant analysis and maximum likelihood linear transformation. For the DNN baseline, speaker adaptation is performed on the features, and nine consecutive frames centered at the current frame are stacked as inputs to the DNN, as specified in the distributed `tri4_nnet` example. Respectively, the two systems achieved PER of 23.8% (GMM) and 22.6% (DNN) without any modification.

We performed a 10-fold cross validation (CV) over the full corpus, by first combining the training and test sets, and creating 10 disparate partitions for each test condition. The gender balance was preserved to be identical to the canonical test set for each test subset, while the phonetic balance was approximately the same but not necessarily identical. This is in order to improve the significance of our PER numbers. The TIMIT corpus is fairly small and the phone occurrence of some phones, or even phone categories, in the test set is lower than ideal. Conducting cross validation on the full set allows us partially address this issue.

For the control experiments of our tests, all configurations of feature extraction and decoding process are retained the same as the baseline. In this case, fair comparisons are guaranteed, and we can fully reveal the effects of our methods in the acoustic model (AM) scoring process.

3.5 Experimental Results

Experimental results examining the two hypotheses proposed above will be presented in this section. We will present the results of over-weighting the landmark frames first. Evaluation of frame dropping will be presented second, and includes several phases. In the first phase, a comparison of different *methods of replacement* is presented, to provide the reader with more insight into these methods before they are applied to acoustic landmarks. In the

[3] `https://github.com/kaldi-asr/kaldi/tree/master/egs/timit/s5`

second phase, we will then leverage our findings to build a strategy that both drops non-landmark frames, and over-weights landmark frames, using the best available *pattern of dropped frames* and *method of replacement*. We open source the code used to carry out the following experiments online. [4]

3.5.1 Hypothesis 1: Over-weighting Landmark Frames

Figure 3.3 illustrates the PER of the strategy of over-weighting the landmark frames during the decoding procedure, and how it varies with the factor used to weight the AM likelihood of frames centered at a landmark. The PER for GMM-based models drops as the weighting factor increases until the factor is 1.5; increasing the weighting factor above 1.5 causes the PER to increase slightly. When the factor is increased to greater than 2.5, the PER increases at a higher slope. Similar trends can be found for DNN models, yet in this case the change in PER is non-concave and spans a smaller range. If landmark frames are under-weighted, or over-weighted by a factor of 1.5 or up to 2.0, PER increases. Over-weighting landmark frames by a factor of 3.0 to 4.0 reduces PER. In this experiment, Wilcoxon tests [90] have been conducted, through Speech Recognition Scoring Toolkit (SCTK) 2.4.10,[5] and tests concluded the difference to be insignificant.

3.5.2 Methods of Replacement for Dropped Frames

Figure 3.4 compares the performance of three *methods of replacement*: `Copy`, `Fill_0` and `Fill_const` when a `Regular` frame dropping pattern is used. Results show that `Fill_0` and `Fill_const` suffer very similar PER increments as the percentage of frames dropped is increased, while `Copy` shows a relatively smaller PER increment for drop rates of 40% or 50%. As for the comparison between acoustic models, DNN-based models outperform GMM-based at all drop rates. Notably, the `Copy` approach synergizes well with DNN models, and is able

[4] `https://github.com/dihe2/kaldi/tree/master/egs/timit/s5`
[5] `https://www.nist.gov/itl/iad/mig/tools`

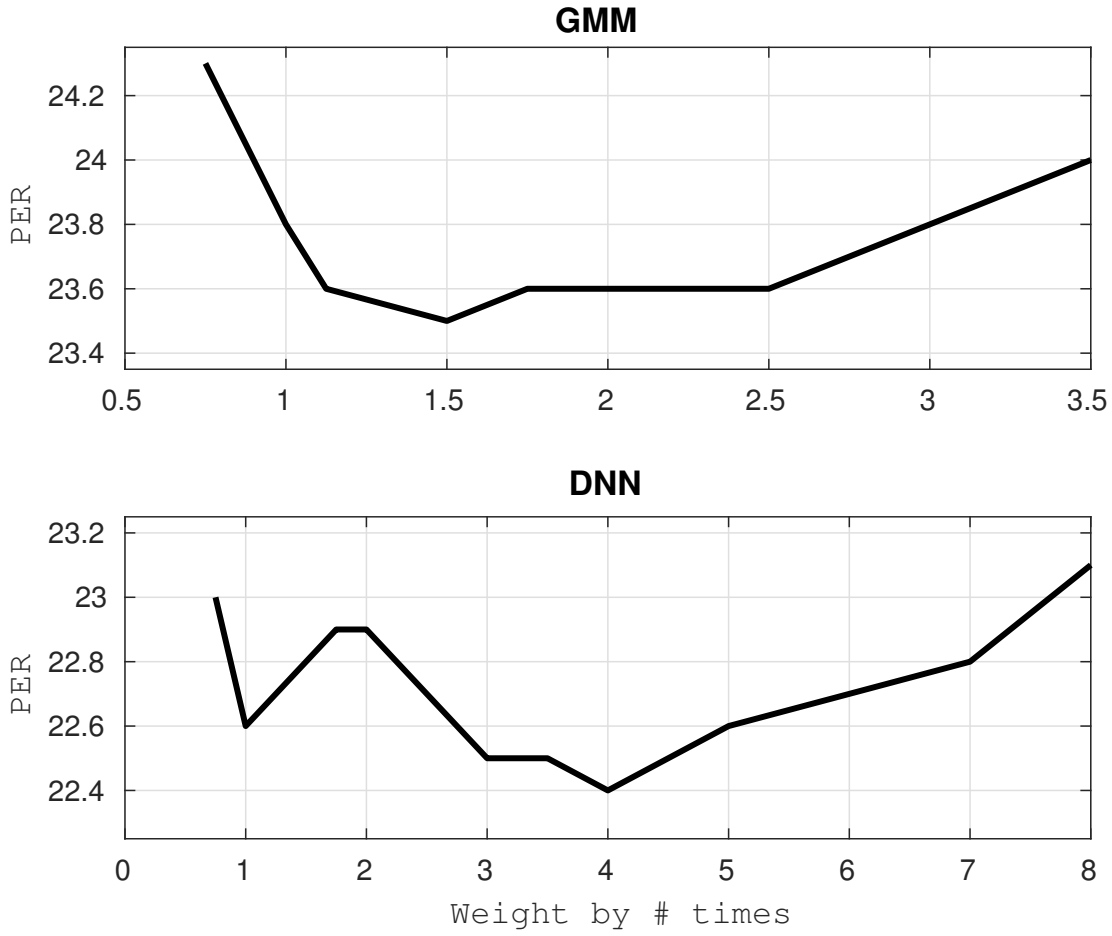


Figure 3.3: Over-weighting landmark frames for GMM and DNN.

to maintain low PER increments even up to 75% drop rate; this finding is similar to findings reported in papers from [50].

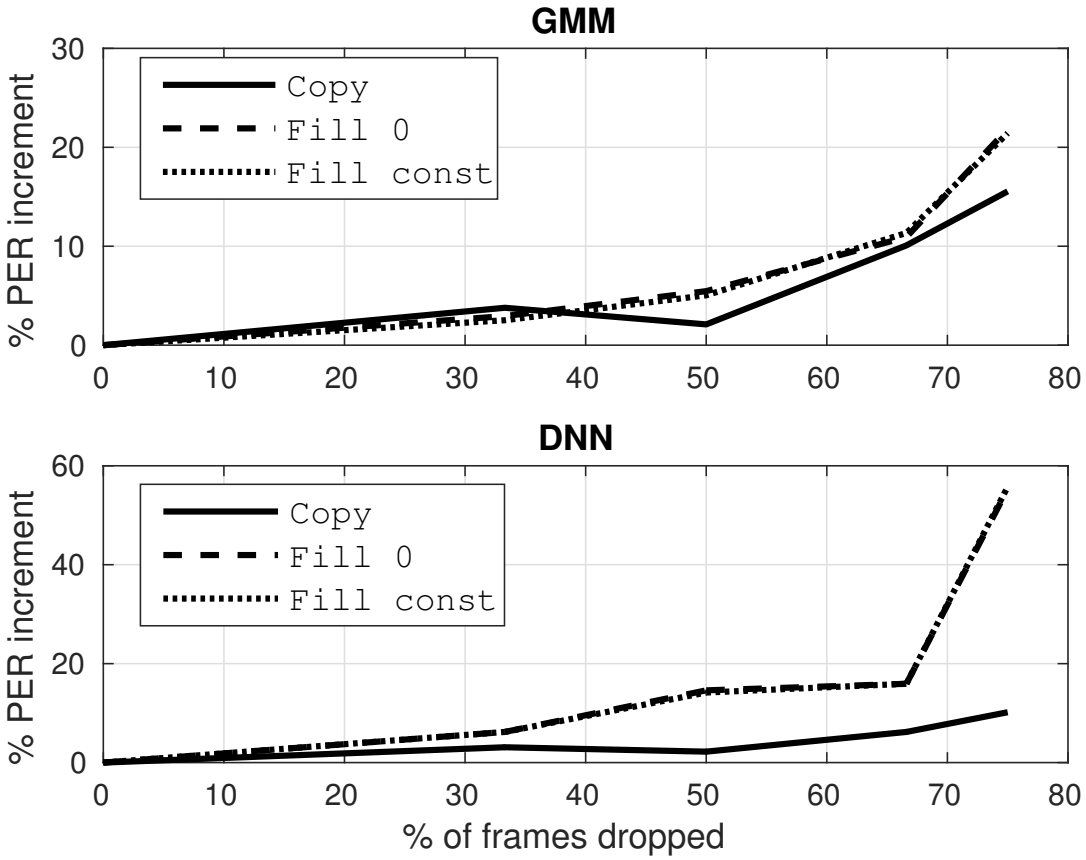


Figure 3.4: Comparison of different methods of frame replacement (`Copy`, `Fill_0` and `Fill_const`) assuming a `Regular` pattern of frame replacement.

Figure 3.5 compares the performance between two *patterns of dropping frames* – `Regular`, `Random`. In both of these the `Copy` method for replacement was used. We also provide for comparison, the `Regular` pattern, but using an `Upsample` replacement method. This scheme uses a 17-tap anti-aliasing FIR filter. The method that offered the lowest phone error rate increment is obtained using a `Regular` pattern with a `Copy` replacement scheme. Results show that `Regular-Copy` outperforms other methods by a large margin in terms of PER increment independent of which AM is used.

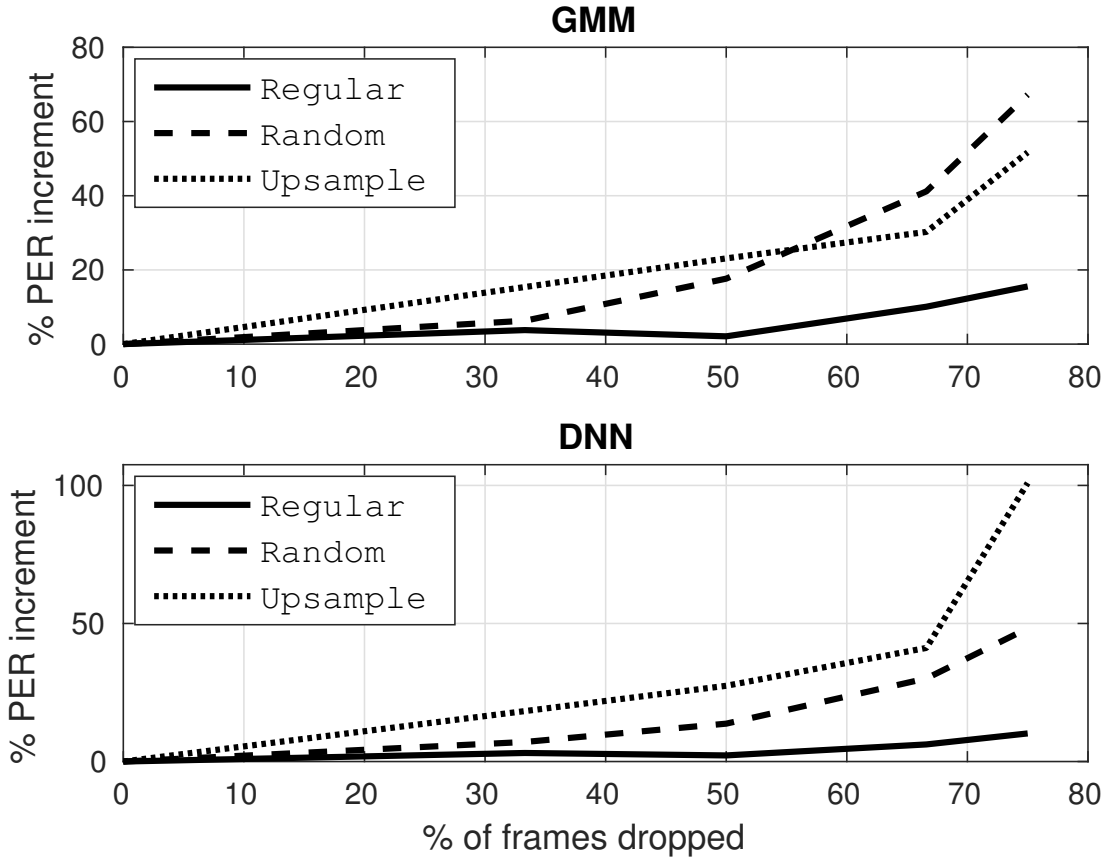


Figure 3.5: Comparison of different patterns of dropping frames assuming `Copy` (`Regular` and `Random`) and `Interpolation through low-pass filtering` (`Upsample`) method of replacement.

3.5.3 Hypothesis 2: Dropping Frames with Regard to Landmarks

At the beginning of this section, experiments that test hypothesis 2 directly are described. The focus is to subject the ASR decoding process to frames missing acoustic likelihood scores, and see how the decoding error rate changes accordingly. Obviously we are interested in using the presence vs. absence of an acoustic landmark as a heuristic to choose the frames to keep or drop. To quantify the importance of the information kept vs. the information discarded, dropping strategies (`Landmark-keep` and `Landmark-drop`) are compared to the non-landmark-based `Random` strategy. Notice the `Regular` strategy has been shown to be more effective than `Random` (e.g., in Fig 3.5); however, to make the PER result meaningful, the same number of frames should be dropped across different patterns being compared. When we keep only landmarks (`Landmark-keep`) or drop only landmarks (`Landmark-drop`), the percentage of frames dropped cannot be precisely controlled by the system designer: it is possible to adjust the number of frames retained at each landmark (thus changing the drop rate), but it is not possible to change the number of landmarks in a given speech sample. Therefore, precisely adjusting the drop rate to meet a different pattern is not practical. Depending on the test set selected, the portion of frames containing landmarks ranges from 18.5% to 20.5%. As opposed to `Random`, `Regular` does not give us the ability to select a drop rate that exactly matches the drop rate of the `Landmark-drop` or `Landmark-keep` strategies. Therefore, it is not covered in the first 2 experiments. However, in the 3rd experiment, we will compare a frame dropping strategy using landmark as heuristic against `Regular` dropping. But that experiment will serve a slightly different purpose.

As in the over-weighting experiment, two types of frame replacement are tested. The `Fill_0` strategy is an exact implementation of hypothesis 2: when frames are dropped, they are replaced by the least informative possible replacement (a log probability of zero). Figure 3.4 shows, however, that the `Copy` strategy is more effective in practice than the

`Fill_0` strategy; therefore these two strategies are tested using a landmark-based frame drop pattern. Figure 3.4 shows that the `Fill_const` strategy returns almost identical results to `Fill_0`, so it is not separately tested here.

Experiment results are presented for both the TIMIT default test split, and for cross-validation (CV) using the whole corpus. The baseline implementation is as distributed with the Kaldi toolkit. Since no frames are dropped, it returns the lowest PER. However, likelihood scoring for the baseline AM will require more computation when compared to a system that drops frames. For CV we report the mean relative PER increment ($\Delta\text{PER} = 100 \times (\text{modified PER} - \text{baseline PER})/(\text{baseline PER})$), with its standard deviation in parentheses, across all folds of CV. Every matching pair of frame-drop systems (`Landmark-keep` versus `Random`) is tested using a two-sample t-test [43], across folds of the CV, in order to determine whether the two PER increments differ. During the t-test, we assume PER numbers from different folds are samples of a random variable. The two-sample t-test intends to find out whether the random variables representing PER for different setups (`Landmark-keep` versus `Random`) have the same mean.

Keeping or Dropping the Landmark Frames

Table 3.2 illustrates the changes in PER increment that result from a `Landmark-keep` strategy (score only landmark frames) versus a `Random` frame-drop strategy set to retain the same percentage of frames. For each test set, we count the landmark frames separately and match the drop rate exactly between the `Landmark-keep` and `Random` strategy. In all cases, the `Landmark-keep` strategy has a lower PER increment. A Wilcoxon test, rather than the two-sample t-test, has been conducted on the default test set; the differences between all pairs but the DNN `Fill0` pair is significant on this test.

For the next experiment we inverted the setup: instead of keeping only landmark frames, we drop only landmark frames (call this the `Landmark-drop` strategy). Tab 3.3 compares the PER increment of a `Landmark-drop` strategy to the increment suffered by a `Random` frame

Table 3.2: PER increments for scoring landmark frames only compared to randomly dropping similar portion of frames (CV stands for cross validation; if the two increments differ, then the lower of the two is marked with either $*$ ($p < 0.05$) or $**$ ($p < 0.001$).)

Acoustic model	GMM				DNN			
Test regime	Default		CV Mean (Stdev)		Default		CV Mean (Stdev)	
Metric	PER (%)	PER Inc (%)	PER (%)	PER Inc (%)	PER (%)	PER Inc (%)	PER (%)	PER Inc (%)
Baseline	23.8	0.0	22.8	0.0	22.7	0.0	20.8	0.0
`Fill_0`								
`Landmark-keep`	36.1	51.7	33.4	46.5(1.34)**	49.6	118.5	49.7	139(10.3)*
`Random`	42.3	77.7	42.1	84.6 (8.35)	50.9	124.2	52.8	154 (14.8)
`Copy`								
`Landmark-keep`	35.2	47.7	32.3	41.5(1.08)**	29.4	29.3	26.9	29.3(0.653)**
`Random`	44.0	84.9	44.1	93.5 (0.734)	38.4	69.3	37.6	80.9 (0.942)

drop strategy with the same percentage of lost frames. The `Landmark-drop` strategy always return higher PER. However, only for the GMM setup `Copy` did we obtain a significant p value during cross validation. The p values for other setups range from 0.13 to 0.17. Again, a Wilcoxon test, rather than the two-sample t-test, has been conducted on the default test set, with the conclusion that only the GMM `Copy` pair demonstrated significant difference.

Table 3.3: PER increments for dropping landmark frames during scoring compared to randomly dropping a similar portion of frames (CV stands for cross validation)

Acoustic model	GMM				DNN			
Test regime	Default		CV Mean (Stdev)		Default		CV Mean (Stdev)	
Metric	PER (%)	PER Inc (%)	PER (%)	PER Inc (%)	PER (%)	PER Inc (%)	PER (%)	PER Inc (%)
Baseline	23.8	0.0	22.8	0.0	22.7	0.0	20.8	0.0
`Fill_0`								
`Landmark-drop`	25.6	7.56	24.0	5.33(1.36)	24.2	6.61	23.1	11.1(1.58)
`Random`	24.1	1.26	23.4	2.68 (1.23)	23.6	3.96	22.4	7.53 (1.24)
`Copy`								
`Landmark-drop`	25.6	7.5	24.1	5.83(0.873)*	24.3	7.1	22.1	6.44(0.836)
`Random`	24.6	3.3	23.1	1.14 (0.948)	23.6	4.0	21.6	3.85 (0.760)

The results in Tab 3.2 demonstrate that keeping landmark frames is better than keeping a random selection of frames at the same drop rate, in all but one of the tested comparison pairs. The results in Tab 3.3 demonstrate that random selection tends to be better than selectively dropping the landmark frames, though the difference is only significant in one of the four comparison pairs. These two findings support the hypothesis that frames containing

landmarks are more important than others. However, the PER increments in some setups are very large, indicating the ASR might no longer be functioning under stable conditions.

Using Landmark as a Heuristic to Achieve Computation Reduction

Methods in Tab 3.2 and 3.3 compared the `Landmark-keep`, `Landmark-drop`, and `Random` frame drop strategies. Table 3.4 illustrates PER increment (%) for the `Landmark-keep` and `Regular` frame-dropping strategies. In this experiment, we are no longer directly testing Hypothesis 2. Instead, we are trying to achieve high frame dropping rate subject to low PER increment. As dropped frames need not be calculated during the acoustic model scoring procedure, a high dropping ratio can benefit the ASR by reducing computational load. The strategy leveraging landmark information is a hybrid strategy: on top of a standard `Regular` strategy, it keeps all landmark frames and over-weights the likelihoods of these frames as in 3.5.1. For each acoustic model type (GMM vs. DNN), three different percentage rates of frame dropping are exemplified. In each case, we select a `Regular` strategy with high dropping rate, modify it to keep the landmark frames, measure the percentage of frames dropped by the resulting strategy, then compare the result to a purely `Regular` frame-drop strategy with a similar drop rate. The baseline `Regular` strategies have three standard drop rates: 33.3% (one out of three frames dropped, uniformly), 50% (one out of two frames dropped), and 66.7% (two out of three frames dropped). Table 3.4 highlights results for one of the setups in bold, as that setup achieves a very good trade-off between high dropping ratio and low PER increment.

As we can see, for DNN acoustic models, the `Landmark-keep` strategy results in lower error rate increment than a `Regular` strategy dropping a similar number of frames. Wilcoxon tests demonstrated a statistically significant difference at all three drop rates. For GMM acoustic models, avoiding landmarks does not seem to return a lower error rate. In fact, the error rate is higher for 2 out of 3 different drop rates. The highlighted case in Tab 3.4 is intriguing because it the PER increment is so low, and this row will therefore serve as the

Table 3.4: PER increments comparison between `Landmark-keep` and `Regular` drop strategies for GMM and DNN.

	Copy	Default		Cross Validation			
		Drop Rate%	PER Inc%	Drop Rate%	PER Inc%	Inc STD%	Inc pVal
GMM	Land	41.0	1.26	44.4	1.84	0.0133	0.962
	Reg	33.3	3.78	33.3	1.81	0.0119	
	Land	54.2	2.94	54.1	2.86	0.0140	0.598
	Reg	50	2.1	50	2.58	0.00780	
	Land	64.3	12.1	65.0	8.10	0.0182	0.159
	Reg	66.7	10.1	66.7	6.91	0.0181	
DNN	Land	41.0	0.44	44.4	1.84	0.0115	0.0011
	Reg	33.3	3.98	33.3	4.20	0.0153	
	Land	**54.2**	**0.44**	**58.4**	**1.90**	**0.167**	**0.0029**
	Reg	50	2.21	50	4.12	0.0115	
	Land	64.2	3.08	69.0	5.86	0.0121	0.0391
	Reg	66.7	6.17	66.7	7.04	0.0160	

Table 3.5: PER increments for Landmark-keeping strategy for DNN with dropping rate near 54.2% and over-weighting factor near 4 times

PER Inc%		Over-weighting Factor		
		3.5	4	4.5
Drop Rate%	52.1	1.42	0.84	0.93
	54.2	0.88	0.44	0.88
	56.3	0.62	0.40	0.40

basis for further experimentation in the next section. In this setup for DNN, over 50% of the frames were dropped, but the PER only increased by 0.44%. This result seems to support the hypothesis that landmark frames contain more information for ASR than other frames, but in Tab 3.4, this row has the appearance of an anomaly, since the error increment is so small. In order to confirm that this specific data point is not a special case, we conducted additional experiments with very similar setups. The results for these additional experiments are presented in Tab 3.5.

Additional results presented in Tab 3.5 are obtained through applying an over-weighting factor close to 4, which is the optimal value found for DNNs in Fig 3.3. The first and third rows in this table randomly keep or drop a small number of non-landmark frames, in order to obtain drop rates of 52.1% and 56.3% respectively. Since the selection is random, multiple

runs of the experiment result in different PER for the same drop rate; therefore we repeated each experiment 10 times and reported the mathematical mean. Since there is a level of randomness in these results, we do not intend to evaluate our hypotheses on these data; rather, the goal of Tab 3.5 is merely to confirm that the highlighted case in Tab 3.4 is a relatively stable result of its parameter settings, and not an anomaly. Since good continuity can be observed across nearby settings, results in Tab 3.5 lend support to the highlighted test case in Tab 3.4.

3.6 Discussion

Results in Sec 3.5.1 tend to support hypothesis 1. However, the tendency is not statistically significant. The tendency is consistent for the GMM-based system, for all over-weighting factors between 1.0 and 3.0. Similar tendencies appeared for over-weight factors between 3.0 and 5.0 for DNN-based system.

Experiments in Sec 3.5.2 tested different non-landmark-based frame drop strategies, and different methods of frame replacement. It was shown that, among the several strategies tested, the `Regular-Copy` strategy obtains the smallest PER. There is an interesting synergy between the frame-drop strategy and the frame-replacement strategy, in that the PER of a 50% `Regular-Copy` system (one out of every two frames dropped) is even better than that of a 33% `Regular-Copy` system (one out of every three frames dropped). This result, although surprising, confirms a similar finding reported by [91]. We suspect that the reason may be relevant to the regularity of the 50% drop rate. When we drop 1 frame out of every 2 frames, the effective time span of each remaining frame is 20ms, with the frame extracted at the center of the time span. Dropping 1 frame out of every 3 frames, on the other hand, results in an effective time span per frame of 15ms, but the alignment of each frame's signal window to its assigned time span alternates from frame to frame.

It is worth mentioning that our definition of acoustic landmarks differs from that of [69] –

specifically, Lulich claims that there is no landmark in the center of Vowel and Glide. Instead, formant-subglottal resonance crossing, which is known to sit between the boundaries of [-Back] and [+Back] vowels, contains a landmark. It is possible that an alternative definition of landmarks might lead to better results.

We can also observe that GMM and DNN acoustic models tend to perform differently in the same setup. For example, for GMM, randomly dropping frames results in a higher PER than up-sampling; this is not the case for DNN models. Results also demonstrate that DNN models perform quite well when frames are missing. A PER increment of only 6% occurs after throwing away 2/3 of the frames. GMM models tend to do much worse, especially when the drop rate goes up.

All experiments on DNN tend to support the strategy to avoid dropping landmarks. However, the 2 test cases covered in Tab 3.3 lack statistical confidence. Scoring only the landmark frames (the `Landmark-keep` strategy) outperforms both `Random` and `Regular` frame-drop-strategies. On the other hand, if landmark frames are dropped (the `Landmark-drop` strategy), we obtain higher PER when compared to randomly scoring a similar number of frames.

We find, at least for ASR with DNN acoustic models, that landmark frames contain information that is more useful to ASR than other frames. In the most striking case, the highlighted result in Tab 3.4 indicates that it is possible to drop more than 54% of the frames but only observe a 0.44% increment in the PER compared to baseline (PER increases from 22.7 to 22.8). We conclude, for DNN-based ASR, that experiments support hypothesis 2 (with statistically significant differences in two out of the three comparisons). In comparison, we failed to find support for hypothesis 2 in GMM-based ASR.

3.6.1 How Landmarks Affect the Decoding Results

Having proven that the `Landmark-keep` strategy is more effective than a `Random` or `Regular` drop strategy, we proceeded to investigate the resulting changes in the rates of insertion,

deletion and confusion among phones. We compared the normalized increment of each type of error, separately, when the confusion matrices of the baseline system are subtracted from the confusion matrices of the `Landmark-keep` and `Random` frame-drop systems. Figure 3.6 compares the normalized error increment, of different types of errors, for the `Landmark-keep` and `Random` strategies. The numbers reported in the figure are normalized error increment. They are calculated using error increment divided by the occurrence of each kind of phone. We use this measure to reflect the increment ratio while avoiding having to deal with situations that could lead to division by zero.

We look into the effect of land-mark based frame dropping in more depth. Figure 3.7 presents the error increment rate table of confusion pairs, insertion and deletion for phones and phones grouped into the types mentioned in Tab 3.1. The numbers reported in the tables are normalized error increment. They are calculated using error increment divided by the phone or phone type occurrence. We use this measure to reflect the increment ratio while avoiding having to deal with situations having to divide-by-0. We compared the error increment rate of dropping all non-landmark frames to randomly dropping frames. The later is set up to drop equal amount of frame as the former. The likelihood scores of the dropped frames have been replaced will a vector filled with 0s. In both tasks, roughly 80% of the frames have been dropped.

Overall, dropping frames causes a minor reduction to the phone insertion rate, while the phone deletion rate significantly worsens. We suspect that after dropping frames, the decoder is less effective at capturing transitions between phones, resulting in correctly detected phones spanning over other phones. In Fig 3.6b we can see that the `Landmark-keep` strategy is more effective than the `Random` strategy, since it returns a lower deletion rate increment. We believe this is because the landmark contains sufficient acoustic information about each phone to force it to be recognized. However, we do not know why the GMM-`Landmark-keep` strategy is less effective at preventing phone deletions than the DNN-`Landmark-keep` strategy. A possible reason might be that more frames were stacked together in the splicing

process for the DNN than for the GMM [85]. If we do consider providing landmarks as extra information to ASR, in order to reduce computation load for example, the difference between GMM and DNN models should be considered.

On the other hand, stacking more feature vectors together does benefit DNN from the confusion perspective, as we can see the confusion pairs error increment rate is marginally lower for the DNN models. We could also see that, in terms of confusion for GMM AM, landmarks do not seem to outperform random on GMM models, as illustrated in Fig 3.7a and 3.8a. Especially for vowels, landmarks actually bring about more confusions than random dropping. This is not the case for DNN models; Fig 3.7c and 3.8c show many deep-blue boxes along the diagonal of the heat map, indicating landmark to be effective at reducing confusion within the same phone type.

When we inspected the phone substitution error information, we found something rather interesting. When only landmark frames have been scored, even though the overall phone confusion increment was small (on average substitution error for DNN increased by 11.2%, for GMM this number increased by 19.3%), we observed dramatic change in substitution count for some phone pairs. For example, there is no confusion from /ɪ/ to /o/ in the baseline setup for both DNN and GMM. However, when we only scored frames containing landmarks, we observed a significant increment in the number of /ɪ/ phones mis-recognized as /o/. For both GMM and DNN, the confusion count from /ɪ/ to /o/ went from 0 to 5. Initially, we suspected the landmark setup failed to recognize /ɪ/ in general. However, we found out that, on average, the confusion count of /ɪ/ decreased by 1.16 for DNN and only increased for 0.0526 for GMM. We examined the distinctive features of /ɪ/ and /o/, however, and found that these two phones have comparable tongue height and sonority, differing only in the features [back] and [constricted_pharynx]. In order to determine whether or not it is true, in general, that the `Landmark-keep` strategy tends to make increased substitution errors for phone pairs with low distinctive-feature distance, we mapped the distinctive features

table[6] for the International Phonetic Alphabet (IPA) into the Carnegie Mellon Pronouncing Dictionary Alphabet[7] using rules listed in IPACMUTI36. Note that the mapping from IPA to CMU is not one-to-one, so we tried to avoid phones that have multiple counterparts in the other alphabet. We found that, many phone pairs with increased substitution errors in the `Landmark-keep` condition were those with similar distinctive features, e.g., /tʃ/ and /t/, /dʒ/ and /z/. Conversely, there are examples of phone pairs with similar acoustics but quite different distinctive features whose substitution error count goes down when the baseline is compared with a `Landmark-keep` strategy. For example, phone /w/ was confused into /v/ in the baseline setup for both GMM and DNN, however, when we only scored the landmark frames, substitution from /w/ to /v/ disappeared. Though both are [+lips], they differ in the feature [consonantal], which is the conditioning feature determining whether or not a number of other features are even labeled; therefore the distinctive feature vectors of these two phones are quite different. However, a number of phone pairs that do not support this finding have been observed, therefore, the tendency cannot be generalized. We do suspect the non-ideal mapping between CMU and IPA might be affecting the results.

Nevertheless, this tendency found for some of the phones confirms that frames extracted at landmarks placed a stronger emphasis on the distinctive features than the baseline. We can see that when we only score landmark frames, the substitution count of some phone pairs seems to be correlated with how similar their distinctive features are. On the other hand, this also exposes a shortcoming of leveraging landmarks heavily. Phones with similar distinctive features might be confused more frequently using a `Landmark-keep` strategy than using a baseline or `Regular` strategy.

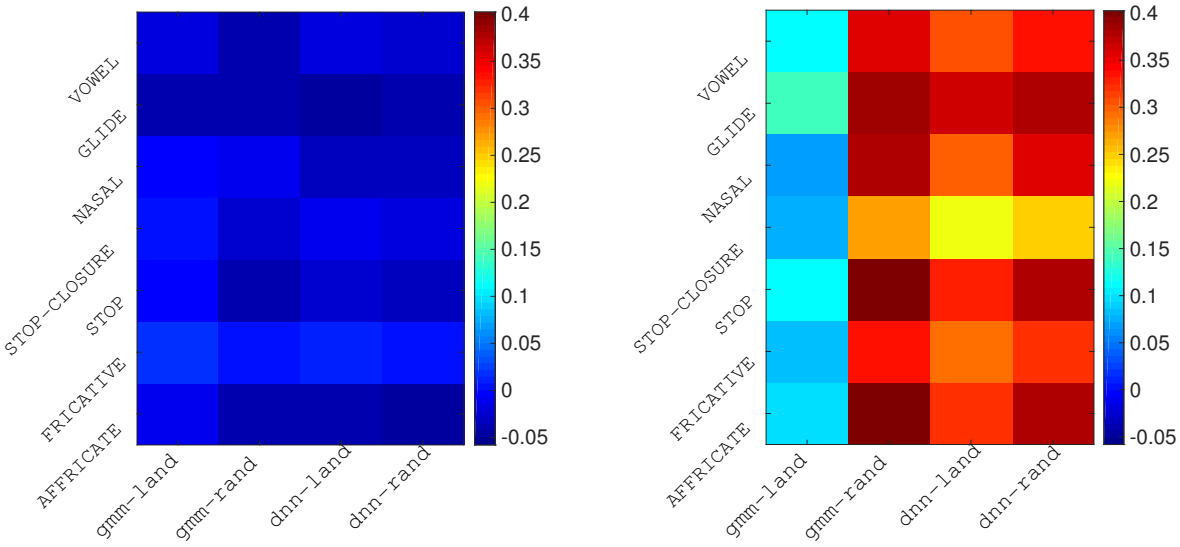
Phone-type-wise, the landmark strategy, compared with random dropping, does seem to perform differently on vowel, stop, fricative and affricative. However, the effect is different for GMM and DNN. While, on average, landmarks reduce the errors for these type of phones

[6] `http://isle.illinois.edu/sst/data/g2ps/English/English_segments.html`
[7] `http://roch.sdsu.edu/cs682/IPA-CMU-TIMIT-Phoneset.pdf`

for DNN, for GMM they actually generate significantly more error. Also landmarks tend to have very different effects on individual phones; as we can see, the outlining boxes in Figs 3.8a and 3.8c are different from their random counterparts. Such effects also vary with AM type; for example, landmarks generate a lot more confusion errors for phone *aw* for GMM, but not for DNN.

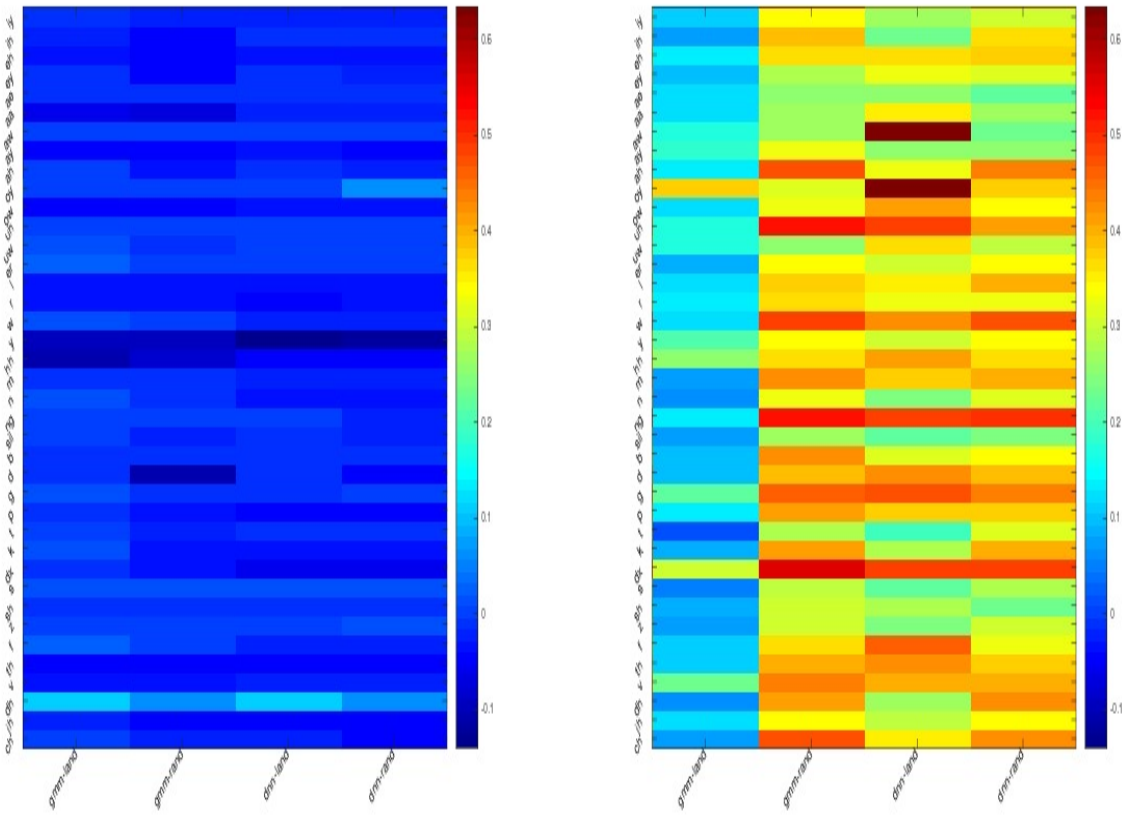
3.7 Recapitulation

Phones can be categorized using binary distinctive features, which can be extracted through acoustic cues anchored at acoustic landmarks in the speech utterance. In this work, we proved through experiments for DNN-based ASR systems operating on MFCC features, on the TIMIT corpus, using both the default and cross validation train-test splits, that frames containing landmarks are more informative than others. We proved that paying extra attention to these frames can potentially compensate for accuracy lost when dropping frames during acoustic model likelihood scoring. We leveraged the help of landmarks as a heuristic to guide frame dropping during speech recognition. In one setup, we dropped more than 54% of the frames while adding only 0.44% to the phone error rate. This demonstrates the potential of landmarks for computational reduction for ASR systems with DNN acoustic models. We conclude that a DNN-based system is able to find a nearly-sufficient summary of the entire spectrogram in frames containing acoustic landmarks, in the sense that, if computational considerations require one to drop 50% or more of all speech frames, one is better off keeping the landmark frames than keeping any other tested set of frames. GMM-based experiments return mixed results, but results for the DNN are consistent and statistically significant: landmark frames contain more information about the phone string than frames without landmarks.



(a) insertion

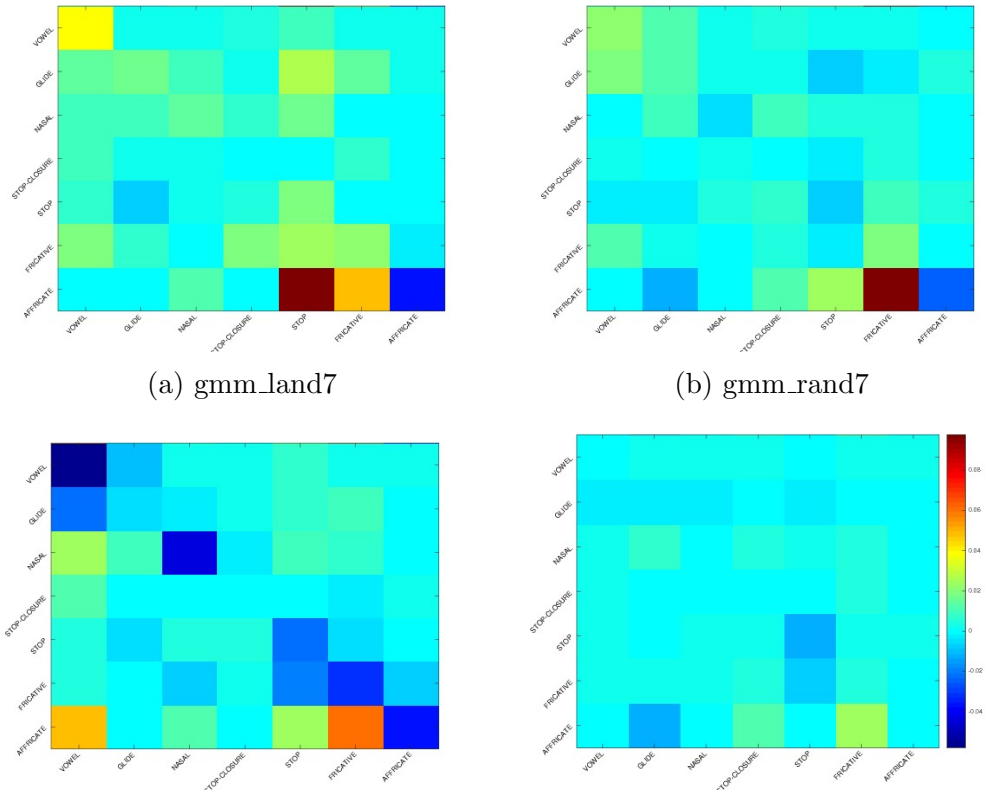
(b) deletion



(c) insertion39

(d) deletion39

Figure 3.6: Insertion and deletion.



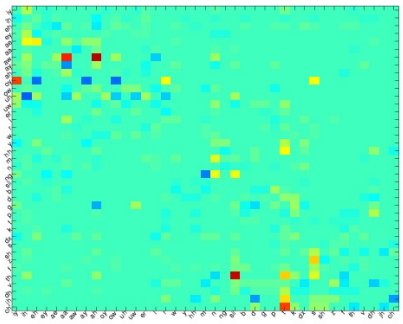
(a) gmm_land7

(b) gmm_rand7

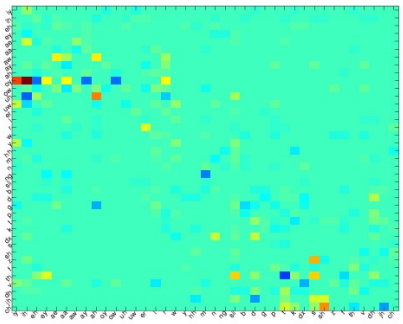
(c) dnn_land7

(d) dnn_rand7

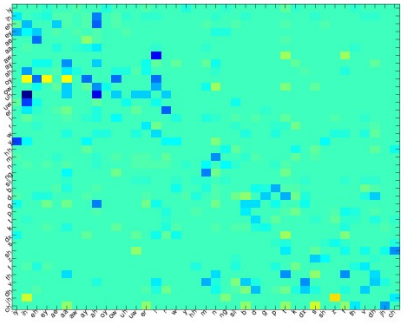
Figure 3.7: Confusion matrices for phone groups.



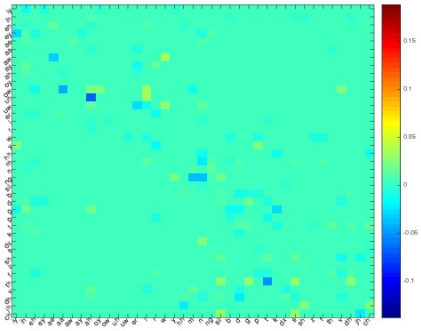
(a) gmm_land39



(b) gmm_rand39



(c) dnn_land39



(d) dnn_rand39

Figure 3.8: Confusion matrices for phones.

Chapter 4

Using Acoustic Landmarks to Improve ASR through MTL

In the early 1980s, Furui [92] demonstrated that the identity of both consonant and vowel can be perceived from a 100ms segment of audio extracted from the C-V transition; in 1985, Stevens [18] proposed that acoustic landmarks are the primary cues for speech perception, and that steady-state regions are secondary or supplemental. Acoustic landmarks produce enhanced response patterns on the mammalian auditory nerve [93], and it has been demonstrated that non-speakers of a language can identify features such as the primary articulator of the landmark [94]. Automatic speech recognition (ASR) systems have been proposed that depend completely on landmarks, with no regard for the steady-state regions of the speech signal [95], and such systems have been demonstrated to be competitive with phone-based ASR under certain circumstances. Other studies have proposed training two separate sets of classifiers, one trained to recognize landmarks, another trained to recognize steady-state phone segments, and fusing the two for improved accuracy [46] or for reduced computational complexity [96, 97]. It has been difficult to build cross-lingual ASR from such systems, however, because very few of the world's languages possess large corpora with the correct timing of consonant release and consonant closure landmarks manually coded. In this chapter we propose a different strategy: we propose to use reference landmark labels in only one language (the source language). A landmark detector trained in the source language is ported to the target language in two ways: (1) by automatically detecting landmark locations in target language test data, and (2) by using landmark detection as a secondary task for the purpose of training a triphone state recognizer that can be more effectively ported cross-lingually. The neural network is trained with triphone state recognition as its primary task;

landmarks are introduced as a secondary task, using the framework of multi-task learning (MTL) [98].

MTL has shown the ability to improve the performance of speech models, especially those based on neural networks [20, 21, 22, 99]. MTL is a mechanism for reducing generalization error. A single-task neural net is provably optimal, for large enough training datasets: as the size of the training dataset goes to infinity, if the number of hidden nodes is set equal to the square root of the number of training samples, the difference between the network error rate and the Bayes error rate goes to zero [100]. MTL is useful when the training dataset is too small to permit zero-error learning [20], or when the training dataset and the test dataset are drawn from slightly different probability distributions (e.g., different languages). In either case, MTL proposes training the network to perform two tasks simultaneously. The secondary task is not important during test time, but if the network is forced to perform the secondary task during training, it will sometimes learn network weights (and consequently, hidden layer activation functions) that are either (1) less prone to over-fitting on the training data than a single-task network, or (2) better generalizable from the distribution of the training data to the distribution of the test data. Landmark detection could potentially be an ideal secondary task for automatic speech recognition (ASR; Fig 4.1), since it detects instantaneous events that are informative to phone recognition. Because landmarks have been demonstrated to correlate with non-linguistic perceptual signals (e.g., enhanced response on the auditory nerve [93]) and because features of a landmark can be classified by non-speakers of the language [94], it is possible that the secondary task of landmark detection and classification will force a neural net to learn weights that are more useful for cross-language ASR adaptation [71] than those of a single-task network. These characteristics are especially helpful for under-resourced languages: in an under-resourced language, training data may be limited, e.g., there may be little or even no transcribed speech. A landmark-based system trained on a well-resourced language might be adapted to an under-resourced language, thus improving ASR accuracy in the under-resourced language. Furthermore, we carried out

experiments reducing the training data in the secondary language, examining the effectiveness of landmark detection as a secondary task for MTL in very low-resourced (40 minutes) scenarios. To our best knowledge, this is the first study where acoustic landmarks have been applied to under-resourced ASR training.

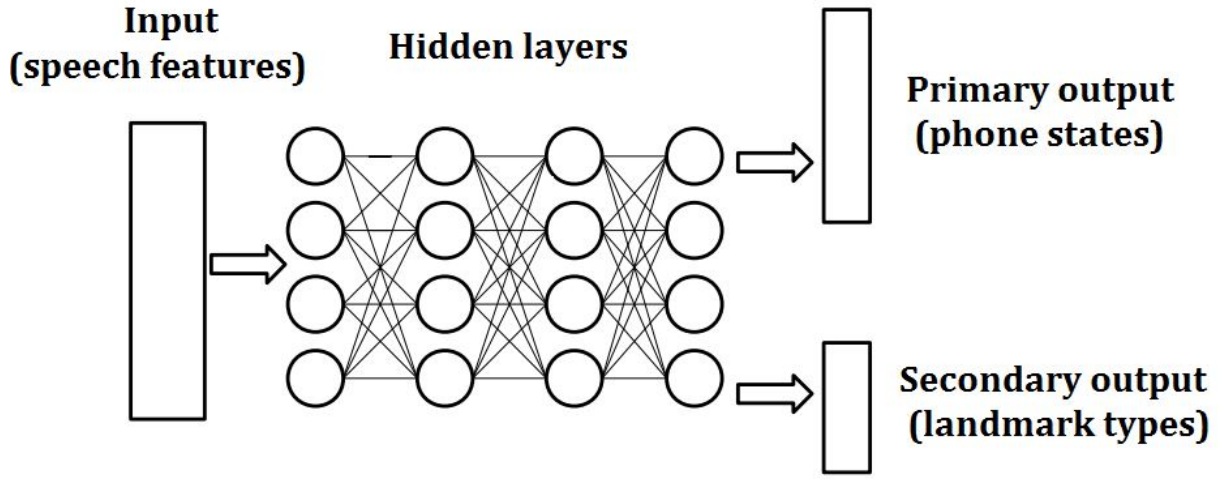


Figure 4.1: MTL neural metwork jointly trained on phone states and landmark types.

The work is presented as follows: After we review some background in Sec 4.1, key methodology and techniques used to apply the landmark theory to MTL are explained in Sec 4.2. Results are presented in Sec 4.3, and the chpter concludes in Sec 4.4.

4.1 Background

Before we talk about our methodology, we would like to briefly review MTL as a neural network training method and talk about the under-resourced corpus we used in this study.

4.1.1 Multi-task Learning

Multi-task learning (MTL) [98] has shown the ability to improve statistical model performance by jointly training a single model for multiple purposes. The multiple tasks in MTL share the same input, but generate multiple outputs predicting likelihoods for a primary

and one or more secondary tasks. When the multiple tasks are related but not identical, or (in the ideal case) complementary to each other, MTL models offer better generalization from training to test corpus [20]. A number of works [20, 21, 22] have proved MTL to be effective on speech processing tasks. Among them [22] proved MTL effective at improving model performance for under-resourced ASR.

When we conduct MTL, for the same input x, we prepare two sets of labels. The label l_i^{ph} specifies the phone or triphone state associated with a frame, while l_j^{la} encodes the presence and type of acoustic landmark. The network is trained in order to minimize, on the training data, a multi-task error metric as shown in Eq 4.1, where $P_i^{ph}(x)$ $(1 \leq i \leq C^{ph})$ is the probability of monophone or triphone state i at frame x as estimated by the neural network, $P_j^{la}(x)$ $(1 \leq j \leq C^{la})$ is the probability of landmark label j at frame x as estimated by the network, and α is a trade-off value we use to weight the two sets of labels. We sweep through a small list of candidate α's to find the value that returns the best result on development test data.

$$\begin{aligned} \mathcal{L}_{mtl} = (1-\alpha) \sum_{i=1}^{C^{ph}} (l_i^{ph} \log(P_i^{ph}(x))) \\ +\alpha \sum_{j=1}^{C^{la}} (l_j^{la} \log(P_j^{la}(x))) \end{aligned} \tag{4.1}$$

4.1.2 The Iban Corpus

The under-resourced language studied in this chapter is Iban [101]. Iban is a language spoken in Borneo, Sarawak (Malaysia), Kalimantan and Brunei. The Malay phone set is similar to English, e.g., the two languages have the same inventory of stop consonants and affricates; Malay also has a relatively transparent orthography, in the sense that the pronunciation of a word is usually well predicted by its written form. If Iban orthography is as transparent as Malay, and if its phone set is as similar to English (an approximated mapping between the

Iban phone set and IPA can be found at github[1]), then it is possible that a landmark detector trained on English may perform well in Iban. However, we are not trying to claim Malay or Iban is a perfect secondary language, when compared to English, for our experiments. These languages are different in many aspects; for example, English in particular is notable for its consonant clustering and use of diphthongs and even triphthongs; this is not the case in Malay. Iban is also selected because of the recent release of an Iban training and test corpus with particularly good quality control [101]. The Iban corpus contains 8 hours of clean speech from 23 speakers. Seventeen speakers contributed $6.8h$ of training data, and the test-set contains $1.18h$ of data from 6 speakers. The language model was trained on a $2M$-word Iban news dataset using SRILM [102]. We foresee that if the primary and under-resource languages share more similarities than English and Iban, we have a good chance of observing better results than what we have obtained.

4.2 Methods

We trained an ASR on the TIMIT corpus using the methods of multi-task learning (Sec 4.1.1), using the detection and classification of landmarks (Sec 4.2.1) as a secondary task. The same ASR is then adapted cross-lingually to the Iban corpus (Sec 4.2.3)

4.2.1 Defining and Marking Landmarks

Landmark definitions in this chapter, listed in Tab 4.1, are based primarily on those of [65], with small modifications. Modifications include the elimination of the +33% and -20% offsets after the beginning or before the end of some phones, reported in [65] and [68], in favor of the simpler definitions in Tab 4.1.

We extracted landmark training labels by referencing the TIMIT human annotated phone boundaries. An example of the labeling is presented in Fig 3.1. This example from [97]

[1] `https://github.com/dihe2/interspeech18/blob/master/phone_mapping.txt`

Table 4.1: Landmark types and their positions for acoustic segments, where 'c', and 'r' denote consonant closure, and release; 'start', 'middle', and 'end' denote three positions across acoustic segments, respectively.

Manner of Articulation	Landmark Type and Position
Vowel	V: middle
Glide	G: middle
Fricative	Fc: start, Fr: end
Affricate	Sr,Fc: start, Fr: end
Nasal	Nc: start, Nr: end
Stop Closure	Sc: start, Sr: end

illustrates the labeling of the word "Symposium".[2] The figure is generated using Praat [103].

Landmarks are relatively infrequent compared to phone-state-labeled speech frames: every frame has a phone label, but fewer than 20% of frames have a landmark label. Because of the sparsity of landmark-labeled frames, we explored different ways to adjust the landmark labels to achieve the best MTL performance. We found, expanding the range of a landmark to include the nearby 2 frames returns the highest accuracy for the primary task.

To further address the imbalance among different landmark classes, the training criterion was computed using a weighted sum of training data, with weights inversely proportion to the class support [104].

4.2.2 Adjusting Landmark Labeling

When applying the landmark labels to MTL, we did encounter difficulties. We failed to realize that our main goal was to train a landmark detector that can effectively compliment the phone state recognizer, not to train a landmark detector that can optimally detect landmark locations. An MTL that over-emphasizes the landmark detection criterion tends to perform poorly as an ASR acoustic model, because landmarks are relatively infrequent compared to phone-state-labeled speech frames: every frame has a phone label, but fewer than 20% of frames have a landmark label. Because of the sparsity of landmark-labeled

[2]selected from file: `TIMIT/TRAIN/DR1/FSMA0/SX361.WAV`

frames, weighting the MTL criterion to emphasize landmark accuracy increased the number of frames receiving the same label, No Landmark, and reduced the benefit of landmark detection as a secondary task for MTL.

We explored different ways to adjust the landmark labels. Table 4.2 covers some of these adjustments. When we label the landmark on only the frame in which it occurs ($\mathbf{ver}_1$), the MTL AM returns high WER. Expanding the range of a landmark to include the nearby 2 frames ($\mathbf{ver}_3$) returned the best result (in comparison, $\mathbf{ver}_2$ only includes the nearby 1 frame). The forth labeling ($\mathbf{ver}_4$) expanded the landmark region, but split the center frame and nearby frames into different classes. The verison $\mathbf{ver}_5$ marked landmark labels similarly to $\mathbf{ver}_4$, but distinctly labeled frames before vs. after the landmark. Expanding the domain of the landmark was helpful ($\mathbf{ver}_3$), but separate classes for frames far from the landmark ($\mathbf{ver}_4$ and $\mathbf{ver}_5$) seemed to be less helpful.

To further address the imbalance among different landmark classes, the training criterion was computed using a weighted sum of training data, with weights inversely proportion to the class support.

Table 4.2: Iban tri-phone WER comparison of different landmark labeling techniques.

Baseline	$\mathbf{ver}_1$	$\mathbf{ver}_2$	$\mathbf{ver}_3$	$\mathbf{ver}_4$	$\mathbf{ver}_5$
18.40	18.31	18.23	18.03	18.16	18.27

4.2.3 Cascading the MTL to Iban

After we trained a landmark detector on TIMIT, we ran the detector on Iban. The English-trained landmark detector output is used to define reference labels for the secondary task of the Iban acoustic model MTL. An example of the detector output on an arbitrary utterance[3] in Iban is given in Fig 4.2. We found that the results are good at outlining fricative landmarks. The detector can also find stop closure landmarks near the correct locations, but with less precision than the fricative landmarks. The performance on vowel and glide

[3] `iban/data/wav/ibm/003/ibm_003_049.wav`

landmarks is only fair: the detector often mixes up the two classes, and incorrectly labels sonorant consonants as vowels.

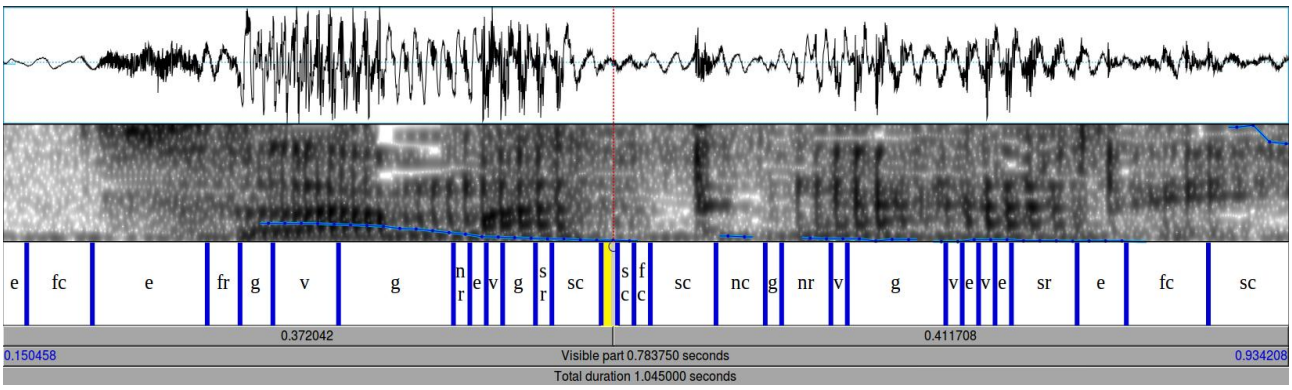


Figure 4.2: Landmark detection result on Iban for utterance ibm_003_049, pronouncing **selamat tengah ari** (**s-aa-l-a-m-a-t t-aa-ng-a-h a-r-i** in Iban phone set). Transcription labels: e=empty (No Landmark); fr, fc, sr, sc, nr, nc, v, g are as in Tab 4.1.

When applying the landmark detector to Iban, we are concerned with the error generated by the detector. The automatically detected landmark labels are treated as ground truth for MTL in landmark-task MTL in Iban; therefore it is possible that erroneously detected landmarks may mislead the network training. To minimize the effect of these mistakes, we introduce an extra weighting factor in the MTL training criterion based on the confidence of the landmark detector output, as shown in Eq 4.2.

$$\begin{aligned}\mathcal{L}_x = (1-\alpha c_x)\sum_{i=1}^{C^{ph}}(l_i^{ph}\log(P_i^{ph}(x))) \\ +\alpha c_x\sum_{j=1}^{C^{la}}(l_j^{la}\log(P_j^{la}(x)))\end{aligned} \tag{4.2}$$

where c_x is a confidence value derived based on the landmark detector output for feature frame x based on Eq 4.3.

$$c_x = P_m^{la\_de}(x) - \frac{1}{C^{la}-1}\sum_{k=1,k\neq m}^{C^{la}}(P_k^{la\_de}(x)) \tag{4.3}$$

where $P_i^{la\_de}(x)$ is the softmax output for landmark class i. The class index $m =_i^{C^{la}} P_i^{la\_de}(x)$, which is also the index for the class the landmark detector predicted.

The intuition behind this extra layer of weighting is to assign a penalty, during training of the ASR, that is proportional to our certainty of its error. If the detector is not confident separating the output class from other classes, then we reduce the loss it generates in the MTL process.

We experimented with multiple ways to initialize the landmark detector and the phone recognizer in the second language. We found that using a network trained through MTL in TIMIT to initialize the MTL network in the second language yields the best results. We found the technique marginally but consistently outperforms other initializations including deep belief networks (DBN) [105].

4.3 Results

All experiments were conducting using the Kaldi [106] toolbox. We extracted an acoustic feature vector using the same algorithm and parameters as [21]. The acoustic model (AM) is a deep neural network with 4 hidden, fully-connected layers, 2048 nodes/layer. The same features and network structure were used for both the landmark detector, the MTL model and the baseline. The baseline is initialized using a DBN [105]. No speaker adaptation is used in any of the ASR systems in this chapter.

Results are reported in Tab 4.3 for both English (TIMIT) and Iban. TIMIT results are reported to indicate the performance of landmark-based MTL in the source language, prior to cross-language adaptation.

On development test sets in both corpora, the value $\alpha = 0.2$ returned the lowest error rate (with little variability in the range $0.1 \leq \alpha \leq 0.3$), and was therefore used for evaluation. For larger α values, such as $\alpha > 0.4$, the WER starts to drop significantly. Error rate higher than the baseline starts to appear, for some setups, when $\alpha \geq 0.6$. The landmark detector

achieves 80.11% frame-wise accuracy in validation. Phone error rate (PER) was reasonably good: 20.6% for the baseline system, and 20.0% for the MTL system, as compared to 22.7% for the open-source Kaldi tri4_nnet recipe.

Decoding results for Iban are reported using word error rate (WER), because the Iban corpus is distributed with automatic but not manual phonetic transcriptions. The comparison between PER in TIMIT and WER in Iban permits us to demonstrate that landmark-based MTL can benefit PER in a source language (English), and WER in an adaptation target language (Iban). Triphone-based ASR trained without MTL on TIMIT, then adapted to Iban, achieves 18.4% WER; a system that is identical but for the addition of landmark-task MTL can achieve 17.93% WER. Neither system includes speaker adaptation, and therefore neither system is better than the 17.45% state of the art WER for this corpus[4] with the same language model.

Table 4.3: Decoding error rate for mono-phone (Mono) and tri-phone (Tri) on TIMIT and Iban.

Corpus	AM	Baseline	MTL	MTL w/ Confid
TIMIT (PER)	Mono	24.6	24.2	NA
	Tri	20.6	20.0	NA
Iban-full (WER)	Mono	24.62	24.22	24.18
	Tri	18.40	18.03	17.93
Iban-25% (WER)	Mono	28.87	27.97	27.64
	Tri	21.31	20.70	20.63
Iban-10% (WER)	Mono	31.16	28.49	28.48
	Tri	25.12	23.64	23.57

As we can see in Tab 4.3, in all cases, regardless of AM and corpus, the ASR system jointly trained with landmark and phone information returns lower error rate. The setups Iban-25% and Iban-10% train the AM on only 25% (100 minutes) and 10% (40 minutes) of the training data uniformly selected at random from the Iban training set (maintaining speaker and gender ratio), but evaluates the error rate on the full test set. As the amount of training data decreases, the benefits of MTL increase. When only 10% of training data

[4] `https://github.com/kaldi-asr/kaldi/blob/master/egs/iban/s5/RESULTS`

is available, simulating a very low resource case, MTL reduces the word error rate by the greatest margin: 8.7% for monophone ASR and 6.17% for triphone ASR. Weighting the MTL loss according to confidence results in a small but consistent error rate reduction. All systems use the same language model, and all systems use acoustic models with the same network architecture and feature set; the error rate change we observe is caused entirely by the use of landmark-task MTL. We foresee that the difference between English and Iban may have some negative effect on the experimental results, and that 2 languages that share more similarities may benefit from our approach even more.

4.4 Recapitulation

This demonstrates that landmark-task MTL results in a neural network that can be more effectively ported cross-lingually. As the amount of training data in the under-resourced language is reduced (from 400 minutes to 100 or 40 minutes), the benefits of landmark-task MTL increase. In addition, introducing a loss weighting according the landmark detector confidence seems to reduce the effect of landmark detector error as it consistently produces lower error rate.

While a cross-language landmark detector provides useful information complementary to the orthographic transcription, visual inspection indicates that a cross-language landmark detector is not as accurate as a same-language landmark detector. Future work, therefore, will train a more accurate landmark detector, using recurrent neural network methods that do not depend on human-annotated phone boundaries, and that can therefore be more readily applied to multi-lingual training corpora.

Chapter 5

Using Acoustic Landmarks to Improve CTC Training through Label Sequence Augmenting

Automatic speech recognition (ASR) is a sequence labeling problem that translates a speech waveform into a sequence of words. Recent success of hidden Markov model (HMM) combined with deep neural networks (DNNs) or recurrent neural networks has achieved a word error rate (WER) on par with human transcribers [1, 2]. These hybrid acoustic models (AMs) are typically optimized by cross-entropy (CE) training which relies on accurate frame-wise context-dependent state alignments pre-generated from a seed AM. The connectionist temporal classification (CTC) loss function [107], in contrast, provides an alternative method of AM training in an end-to-end fashion—it directly addresses the sequence labeling problem without prior frame-wise alignments. CTC is capable of learning to construct frame-wise paths implicitly bridging between the input speech waveform and its context-independent target, and it has been demonstrated to outperform hybrid HMM systems when the amount of training data is large [10, 108, 12]. However, its performance degrades and is even worse than traditional CE training when applied to small-scale data [109].

Training CTC models can be time-consuming and sometimes models are apt to converge to even a sub-optimal alignment, especially on resource-constrained data. In order to alleviate such common problems of CTC training, additional tricks are needed, for example, ordering training utterances by their lengths [12] or bootstrapping CTC models with models CE-trained on fixed alignments [110]. The success of bootstrapping with prior alignments indicates that external phonetic knowledge may help to regularize CTC training towards stable and fast convergence. Furthermore, another investigation [111] reveals that the spiky predictions of CTC models tend to overlap with the vicinity of acoustic landmarks where

abrupt manner changes of articulation occur [56]. The possible coincidence of CTC peaks overlapping acoustic landmarks suggests a number of possible approaches for reducing the data requirements of CTC, including cross-language transfer (using the relative language-independence of acoustic landmarks [112]) and informative priors.

Many efforts have been made to augment acoustic modeling with acoustic landmarks [112, 96, 97] which are detected by accurate time-aligned phonetic transcriptions. To the best of our knowledge, only TIMIT [83] (5.4 hours) provides such fine-grained transcriptions. The value of testing these approaches is limited since the only available corpus is very small. It is worth further exploring the power of landmark theory when scaled up to large corpus speech recognition.

In this chapter, we propose to augment phone sequences with acoustic landmarks for CTC acoustic modeling and leverage a two-phase training procedure with pretraining and finetuning to address CTC convergence problems. Experiments on TIMIT demonstrate that our approaches not only help CTC models converge more rapidly and smoothly, but also achieve a lower phone error rate, up to 8.72% phone error rate reduction over CTC baseline with phone labels only. We also investigate the sensitivity of our approaches to the size of training data on subsets of TIMIT (smaller corpora) and WSJ [113] (a larger corpus). Our findings demonstrate that label augmentation generalizes to larger and smaller training datasets, and we believe this is the first work that applies acoustic landmark theory to a mid-sized ASR corpus.

5.1 Background

5.1.1 Connectionist Temporal Classification (CTC)

Recent end-to-end systems have attracted much attention, for example, because they avoid time-consuming iterations between alignment and model building [107, 114]. The CTC loss computes the total likelihood of the target label sequence over all possible alignments

given an input feature sequence, so that the computation is more expensive than frame-wise cross-entropy training. A blank symbol is introduced to compensate for the difference in length between an input feature sequence and its target label sequence. Forward-backward algorithms are used to efficiently sum the likelihood over all possible alignments. The CTC loss is defined as

$$\mathcal{L}_{ctc} = -\log p\left(\mathbf{y}|\mathbf{x}\right) = -\log \sum_{\boldsymbol{\pi} \in \mathcal{B}^{-1}(\mathbf{y})} p(\boldsymbol{\pi}|\mathbf{x})$$

where $\mathbf{x}$ is an input feature sequence, $\mathbf{y}$ is the target label sequence of $\mathbf{x}$, $\boldsymbol{\pi}$ is one of blank-augmented alignments of $\mathbf{y}$, and $\mathcal{B}^{-1}(\mathbf{y})$ calculates the set of all such alignments. During decoding, the n-best list of predicted label sequences can be achieved by either a greedy search or a beam search based on weighted finite state transducers (WFSTs). In the following experiments, our acoustic models are trained by the phoneme CTC loss, and we report phone error rates on TIMIT (a smaller corpus) through an one-best greedy search and word error rates on WSJ (a larger corpus) through an one-best WFSTs beam search, respectively.

A hybrid neural net-hidden Markov model (NN-HMM), represented by that trained by Kaldi [106], is usually trained on the level of context-dependent sub-phone units, for example, tied tri-phone states. Unfortunately, the mapping between AM states and language model states is not one-to-one. In order to map from AM states to language model states, NN-HMMs must learn and then store a set of state-mapping tables, possibly in the form of weighted finite state transducers. CTC training [107] incorporates all state mapping into a single learning process, reducing the number of incompatibly formatted data tables that must be learned and stored, and the number of steps one has to go through to train them. That said, CTC models are not without shortcomings.

CTC suffers from long training time to converge and requires a large amount of training data, especially when the neural networks are deep [91, 115]. Even for corpora with size over 100 hours, such as Wall Street Journal (WSJ) [113], CTC models under-perform hybrid

HMM systems [108].

The key difference of CTC compared to an NN-HMM is that the latter requires target labels for every frame, whereas CTC computes $p(y|x)$ through accommodating $\boldsymbol{\pi}$ that satisfy $\boldsymbol{\pi} \in \mathcal{B}^{-1}(\mathbf{y})$. The mapping function $B(\cdot)$ is many-to-one.

As a result, the loss calculation process needs to exhaustively enumerate all possible paths that map to the same label sequence $\mathbf{y}$. Despite efforts to speed up this search process, these models still suffer from longer training time compared to DNN-HMM hybrid models.

5.1.2 Acoustic Landmarks

Acoustic landmark theory originates from experimental studies of human speech production and speech perception. It claims there exist instantaneous acoustic events that are perceptually salient and sufficient to distinguish phonemes [56]. Automatic landmark detectors can be knowledge-based [65] or learned [46]. Landmark-based ASR has been shown to slightly reduce the WER of a large-vocabulary speech recognizer, but only in a rescoring paradigm using a very small test set [46]. Landmarks can reduce computational load for DNN/HMM hybrid models [96, 97] and can improve recognition accuracy [112]. Previous works [112, 96, 97, 116] annotated landmark positions mostly following experimental findings presented in [117, 68]. Four different landmarks are defined to capture positions of vowel peak, glide valley in glide-like consonants, oral closure and oral release.

5.2 Methods

5.2.1 Distinctive Features and Landmark Definition

Distinctive features (DFs) concisely describe sounds of a language at a sub-segmental level, and they have direct relations to acoustics and articulation. These features take on binary encodings of perceptual, phonological, and articulatory speech sounds [118]. A collection

of these binary features can distinguish each segment from all others in a language. Autosegmental phonology [119] also suggests that DFs have an internal organization with a hierarchical relationship with each other. We follow these linguistic rules to select two primary features—*sonorant* and *continuant*—that distinguish among the manner classes of articulation, resulting in a four-way categorization shown in Tab 5.1. We define landmarks to be changes in the value of one of these two distinctive features using the TIMIT phone inventory.

The standard phoneme set used by WSJ ignores detailed annotations of oral closures, for example /bcl/, so that we merge together [-,+*continuant*] features under [-*sonorant*] column in Tab 5.1, resulting in a three-way categorization for WSJ experiments instead.

Table 5.1: Broad classes of sounds on TIMIT.

Manner	-sonorant	+sonorant
-continuant	bcl dcl gcl kcl pcl q tcl	em en eng m n ng
+continuant	b d g k p t ch jh dh f hh hv s sh th v z zh	aa ae ah ao aw ax ax-h axr ay dx eh el ey ih ix iy l nv ow oy r uh uw ux w y er

5.2.2 Augmenting Phone Sequences with Landmarks

We defined two methods of augmenting phone label sequences with acoustic landmarks. *Mixed Label 1* only inserts landmarks between two broad classes of sounds where manner changes occur; *Mixed Label 2* inserts landmarks between phones even if manner changes don't exist. Figure 5.1 demonstrates an example of our two augmentation methods.

CTC only requires a single target label sequence, so that augmenting phone sequences with landmarks can relax the need for time-aligned phone transcriptions. With a blank label present between two phones in the training target sequence, the vanilla CTC training can be considered as already experimenting with the scenario where a dedicated phone boundary

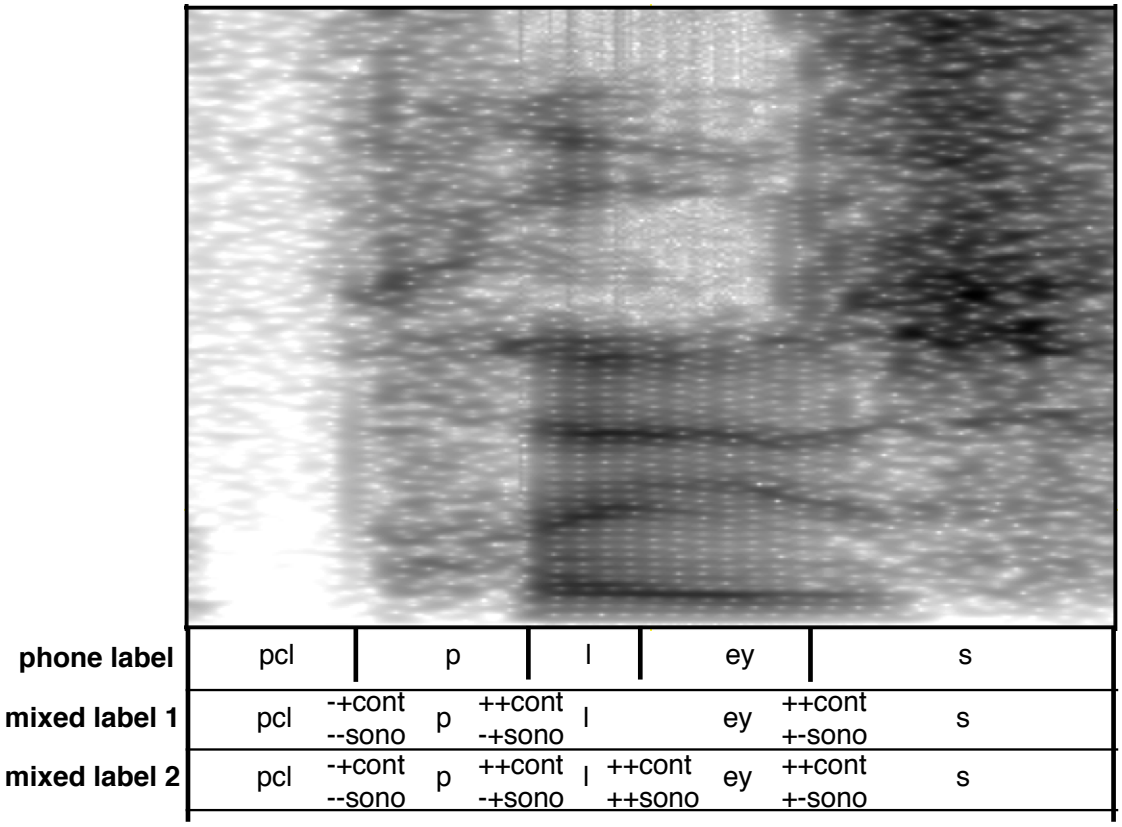


Figure 5.1: Examples of target label sequences for the word "PLACE". The audio clip is selected from `SI792` on TIMIT.

label is added to the label set. CTC is thus an ideal baseline for our experiments.

As mentioned in Sec 5.2.1, we designed two mixed labels with manner changes. Figure 5.1 illustrates the details of our annotations.

Depending on the labeling methods selected as mentioned in Sec 5.2.1, we might add a landmark label between two phones according to the respective manner changes between the phones. Take the word lag (**l**, **ae**, **g**) for example, the new mixed label sequence will become **l**, **ae**, **cont+sono+** $\Rightarrow$ **cont+sono-**, **g** if we use the first labeling method and **l**, **cont+sono+** $\Rightarrow$ **cont+sono+**, **ae**, **cont+sono+** $\Rightarrow$ **cont+sono-**, **g** if we use the second method. In the above example, **cont+sono-** and **cont+sono+** represent phones falling into the bottom left and right category of Tab 5.1.

5.2.3 Acoustic Modeling using CTC

We follow a pretraining and finetuning procedure to train our CTC models. At the phase of pretraining, the AM initializes weights randomly and is trained by one of our mixed label sequences until convergence; at the phase of finetuning, the AM initializes weights from the pretrained model and continues to be trained by a label sequence with only phones. These two phases of training take the same acoustic features. Figure 5.2 briefly illustrates the whole procedure. The top output layer calculates a posterior distribution over symbols combined with both phones and landmarks, while the bottom output layer calculates it over only phones.

5.3 Experiments

5.3.1 Configurations

We conducted our experiments on both the TIMIT [83] and WSJ [113] corpora. We used 40-dimensional log mel filterbank energy features computed with 10ms shift and 20ms span.

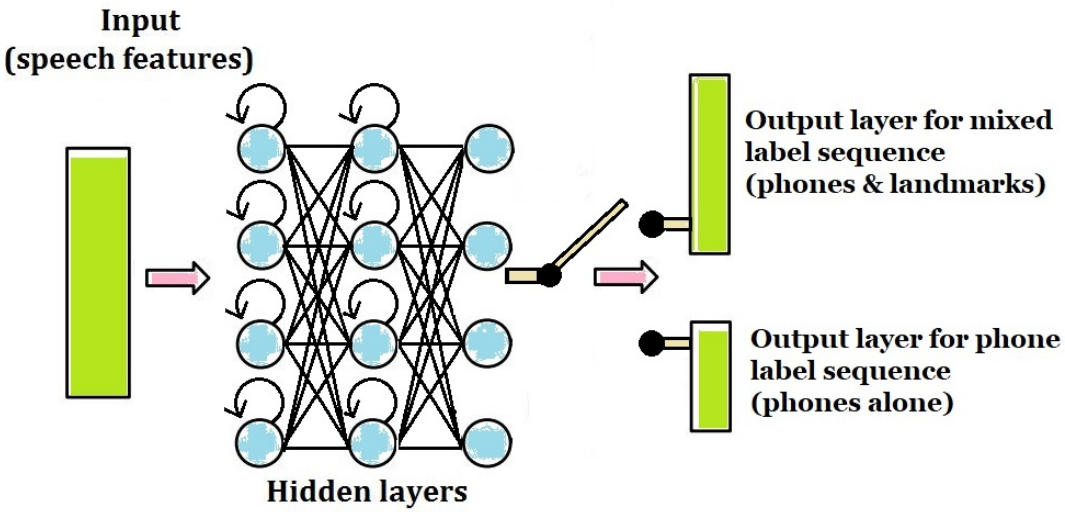


Figure 5.2: Two-phase acoustic modeling: top output layer pretrains with mixed labels and bottom output layer finetunes with phone labels only.

No delta features or frame stacking were used. The recurrent neural networks stacked two layers of bidirectional LSTMs, each with 1024 cells (512 cells per direction), capped by a fully connected layer with 256 neurons. Weights are initialized randomly from Xavier uniform distribution [120]. New-Bob annealing [121] is used for early stopping after a minimum waiting period of two epochs. The initial learning rate is 0.0005. The TIMIT baseline is trained on 61 phones. The WSJ baseline is trained on 39 phones[1] defined in the CMU pronunciation dictionary. One-best greedy search is applied to calculate the phone error rate (PER). We did not map TIMIT phones to CMU phone set (39 phones). In order to make a fair comparison, all baselines went through the same two-phase training with pretraining and finetuning. One-best beam search based on WFSTs is applied to calculate the word error rate in WSJ experiments using decoding graphs with a primitive trigram (tg) and pruned trigram (tgpr) from EESEN.[2] We use the same train/dev/test split from Kaldi Recipes for TIMIT and WSJ.

[1] `https://github.com/Alexir/CMUdict/blob/master/cmudict-0.7b.phones`
[2] `https://github.com/srvk/eesen/blob/master/asr_egs/wsj/run_ctc_phn.sh`

5.3.2 Experiments on TIMIT

Figure 5.3 presents the development set PER as a function of training epoch. The PER for mixed sequence represented by the red and yellow lines in Fig 5.3 is calculated after landmark labels have been removed from the output sequence. In the pretrain phase, models trained on augmented labels do not seem to have any advantage in terms of error rate. However, the models converge much more rapidly and smoothly. After pretraining, both the baseline and mixed-label systems are finetuned; the mixed-label system (purple line in Fig 5.3) returns a model that is more accurate.

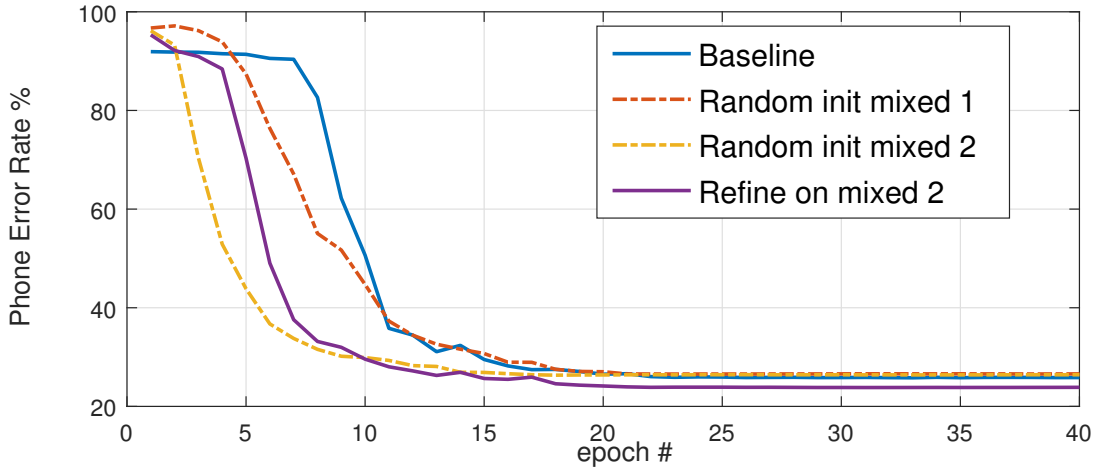


Figure 5.3: PER as a function of training epoch. PER is calculated against only phones after landmarks are removed.

The exact PERs for different setups on the TIMIT test set are reported in Tab 5.2. Our baseline achieved a PER of 30.36%, which was not improved by finetuning. This is higher than PER reported elsewhere (e.g., [107]), because nobody else calculates PER on the full TIMIT set of 61 phones. As shown in Tab 5.2, if we train with mixed labels and strip away landmarks from the hypothesis sequence, landmarks provide little benefit. However, the *Mixed 1* and *Mixed 2* systems achieved lower PER after the finetuning stage by 4.64% and 8.72% relative, respectively. Apparently, a phone sequence augmented with landmarks can be learned more accurately than a raw phone sequence, perhaps because the acoustic

features of manner transitions are easy to learn, and help to time-align the training corpus. The *Mixed Label 2* set outperforms *Mixed Label 1*, apparently because the extra boundary information in *Mixed Label 2* is useful to the training algorithm.

Table 5.2: Comparison between baseline and our proposed models with augmented target labels in PER (%). Number in the parentheses denotes the relative reduction over baseline.

	Baseline	Mixed 1	Mixed 2
random init	30.36	30.98	29.10
finetuned	30.36	**28.96** (4.64%)	**27.72** (8.72%)

It is not clear why a finetuning stage is needed in order for *Mixed 1* to beat the baseline. One possibility is that landmark labels are helpful for some tokens, and harmful for others; pretraining uses the helpful landmarks to learn better phone alignments, then finetuning permits the network to learn to ignore the harmful landmark tokens. We looked into the prior distribution on TIMIT, presented in Fig 5.4, of both phones (top subplot, with phones ordered in the same way as they occurred in Tab 5.1) and landmarks (bottom subplot, *Mixed Label 2* ordered in category permutation using *continuant* as the first variable and *sonorant* as the second). The table reveals that the distribution of landmarks is not balanced. Most labels indicate a transition related to the [*+continuant,+sonorant*] phones. A skewed landmark support is not ideal for augmenting phone recognizer training as it tends to provide the same and redundant information for many training sequences.

5.3.3 Datasets Smaller and Larger than TIMIT

To confirm our findings, we further investigated the sensitivity of our approaches to the size of training data on subsets of TIMIT (smaller corpora) and WSJ (a larger corpus). In this section, we only demonstrate the experiments using *Mixed Label 2* augmentation method since it outperforms *Mixed Label 1* in the previous discussion. We report PER/WER results for finetuned models.

Figure 5.5 shows the PER results by stretching the amount of training data on TIMIT.

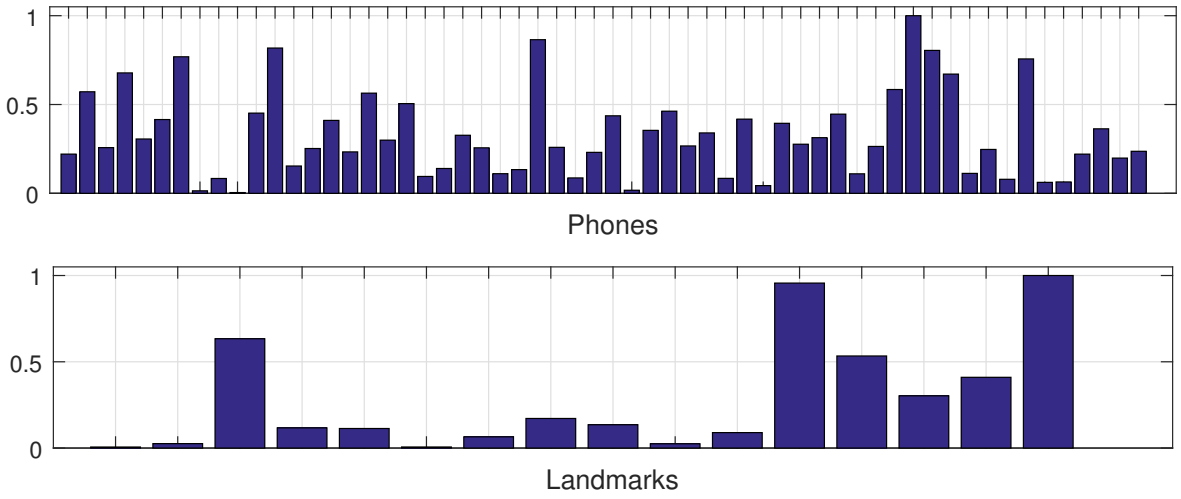


Figure 5.4: Prior distributions of phones and acoustic landmarks.

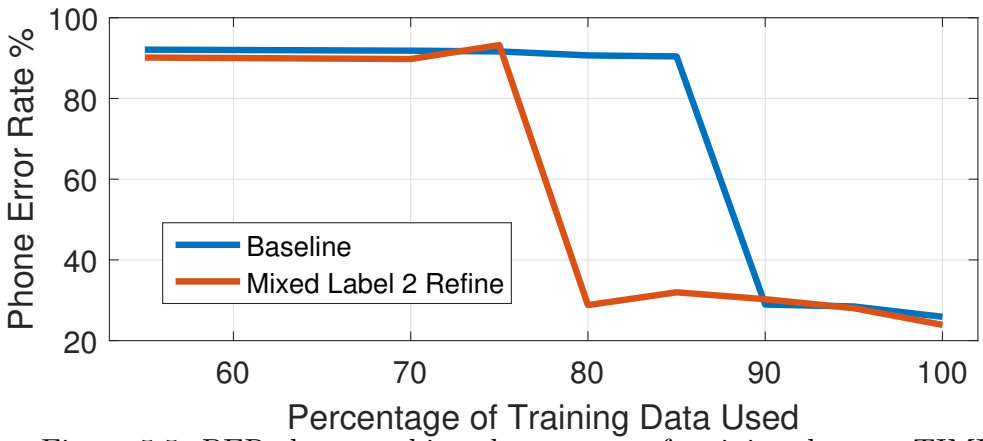


Figure 5.5: PERs by stretching the amount of training data on TIMIT.

Both the proposed model and baseline fail to converge when 75% of the training data is used. We observe that both models start to predict a constant sequence (usually made up of two to three most frequent phones) for all utterances. Scheduled reducing the learning rate by New-Bob annealing cannot help to converge to an optimal. Increasing the amount of training data helps both models converge. The baseline needs 90% of TIMIT to converge, while the proposed system only needs 80% of TIMIT.

When scaling up to a even larger corpus on WSJ, the proposed *Mixed Label 2* system could achieve better performance over the baseline consistently in terms of all metrics as shown in Tab 5.3. Our baseline system slightly under-performs the results published in EESEN [108] because our network is shallower and the acoustic inputs do not include any dynamic (delta) features, but the benefit of the proposed landmark augmentation method still applies. To our knowledge, this is the first work to show that manner-change acoustic landmarks reduce both PER and WER on a mid-sized ASR corpus.

Table 5.3: Label error rate (%) on WSJ, where tg and tgpr denote decoding graphs with primitive and pruned trigrams.

	PER		WER (tgpr / tg)	
	eval92	dev93	eval92	dev93
Baseline	8.7	12.38	8.75/8.17	13.15/12.31
Mixed 2	**8.12**	**11.49**	**8.35**/8.19	**12.86/12.28**

5.4 Recapitulation

We proposed to augment CTC with acoustic landmarks. We modified the classic landmark definition to suit the CTC criterion and implemented a pretraining-finetuning training procedure to improve CTC AMs. Experiments on TIMIT and WSJ demonstrated that CTC training becomes more stable and rapid when phone label sequences are augmented by landmarks, and achieves a significantly lower (8.72% relative reduction) asymptotic PER. The advantage is consistent across corpora (TIMIT, WSJ) and across metrics (PER, WER).

CTC with landmarks converges when the dataset is too small to train the baseline, and it also converges without the need of time alignments on a mid-sized standard ASR training corpus (WSJ).

Chapter 6

Conclusion and Future Work

In this study, we attempt to introduce audio perception theories for the aid of audio processing systems. Audio perception theories, in this study, specifically auditory roughness and acoustic landmark, came from a more science-oriented background where human perception and acoustic articulation are the focus of the study. In contrast, research on audio processing systems originated from the need for practical applications (AED and ASR) and is much more engineering focused. A study that bridges the two sub-areas of audio research benefits both sides. The findings of this study are summarized in Sec 6.1.

6.1 Summary of Key Contributions

A list of findings resulting from this study follows. These findings have been published in multiple conferences and journals [122, 96, 123, 97, 112, 124].

6.1.1 List of Contributions

- Found experimental evidence that Auditory Roughness (AR) can serve as a pre-filtering feature for AED systems targeting screaming or human affective speech.
- Developed a low complexity approximation to the classic AR feature that provides comparable discriminate capability but only requires computational load similar to STE to extract.
- Demonstrated the approximated AR on a low-pow, low-cost FPGA-based multi-microphone,

wireless AED system. This system won the first Hardware Design Contest held by DAC 2017.

- Found experimental evidence that acoustic landmarks can improve DNN-based AM accuracy through over-weighting landmark frames during AM likelihood inference.
- Designed a strategy that combines over-weighting and dropping likelihood of feature frames to reduce the DNN-based AM inference computational load by over 54% while taking very minor, 0.44% relatively, accuracy lost.
- Backed the frame over-weighting and dropping experiment with rich experiments and significant tests, solidly confirming the effectiveness of landmarks on DNN-based AM.
- Augmented DNN-based AM training with acoustic landmark information and observed model error rate reduction.
- Migrated Acoustic Landmark detector trained in a resource-rich language to a secondary language and found experimental evidence that shows the output of landmark detector can be used as MTL labels; landmark detector training in one language can be used to benefit a different language for DNN-based AM training.
- Developed a pre-train and finetuning strategy to leverage acoustic landmark augmented label sequence to train an end-to-end AM with CTC loss.
- For the first time, found experimental evidence that acoustic landmark can benefit non-frame synchronized AM (CTC-based) and presented supportive results on a mid-size LVCSR corpus (WSJ).
- Found experimental evidence that acoustic landmark augmented CTC models have better convergence characteristics when training data is limited.

6.1.2 Connecting the Findings

The first three contributions listed above can be summarized in Fig 6.1.

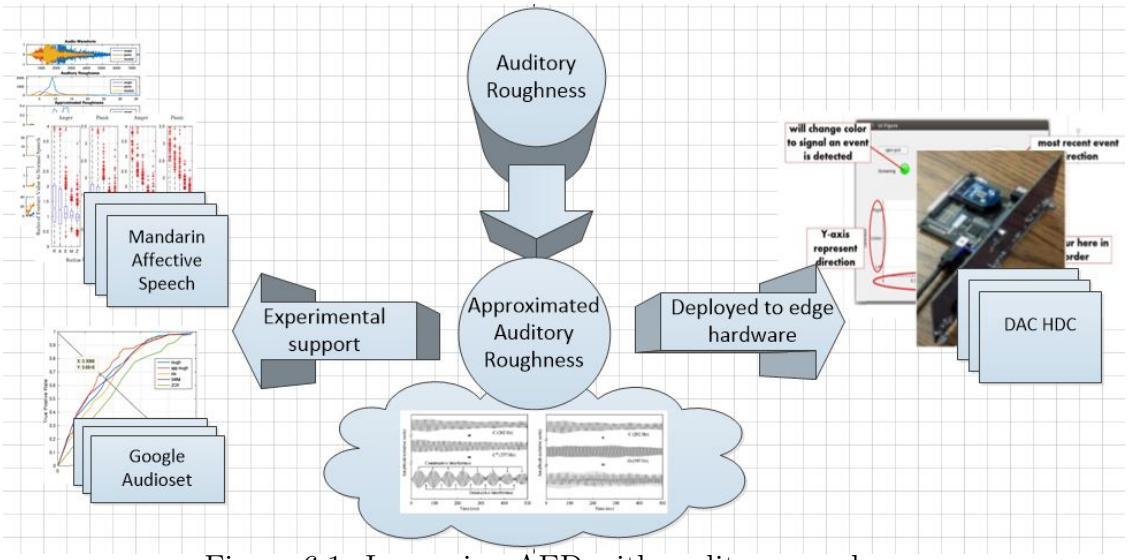


Figure 6.1: Improving AED with auditory roughness.

The studies related to acoustic landmarks can be summarized by Fig 6.2. The left of the figure expresses the relationship between the studies on frame-synchronized ASR systems while the right side covers works on end-to-end systems. The top left part of Fig 6.2 focuses on experimenting with the feature frames assuming a pre-trained AM, which refers to the 4th to 6th points mentioned in Sec 6.1.1. The bottom left part considers re-training an AM with the help of acoustic landmarks. This is covered by the 7th and 8th point in Sec 6.1.1.

6.2 Future Work

Even though the work on auditory roughness is concluded, work on applying acoustic landmark to ASR still faces many open questions. Preliminary results presented in Chapter 3 and 4 lend support to the hypothesis that acoustic landmarks can potentially augment ASR systems. However, the results are not strong and convincing enough. The most significant weakness of these results is that the same findings have not been observed on a larger speech corpus. Obviously, repeating the work in Chapter 4 on more corpora from more languages

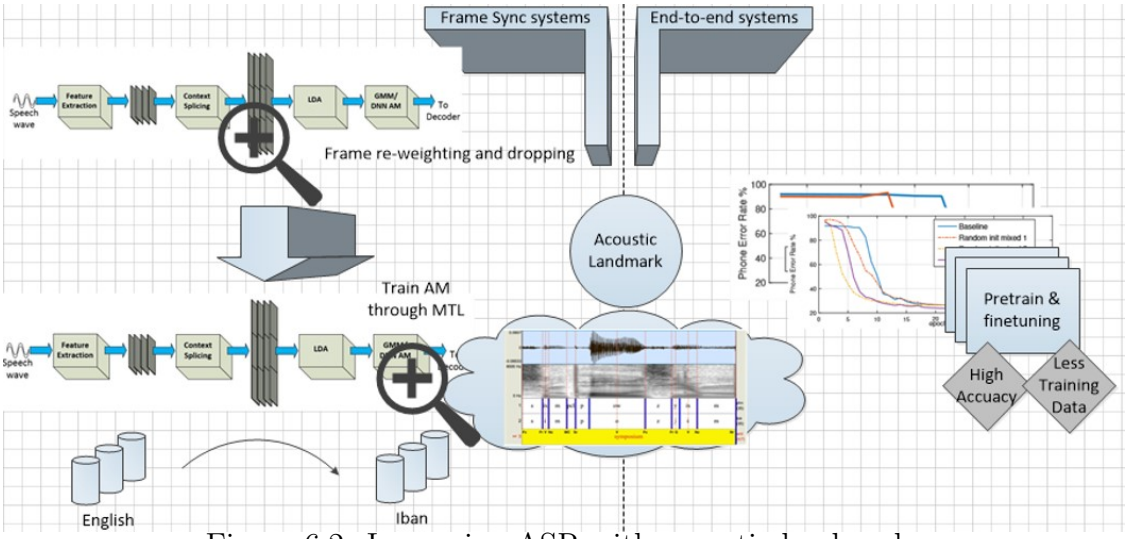


Figure 6.2: Improving ASR with acoustic landmarks.

will provide a thorough test. However, the same cannot be said for Chapter 3. This is, mainly due to the limitation on speech corpus. Chapter 3 claims that if we are able to accurately detect acoustic landmark, we can benefit the ASR. Yet considering the landmark detection reported in Sec 3.1, there really is a significant gap between machine detected landmarks and landmarks derived though human annotated phone boundaries. A better landmark detector can also benefit MTL on a different corpus or language. Reducing detection error can potentially close the gap between error rate reduction on the native corpus (TIMIT) and the test corpus (Iban or other corpus).

6.2.1 A Better Acoustic Landmark Detector

As we can see in Fig 4.2, the landmark detection on Iban is far from perfect. It is questionable whether forced-aligned results can be used to cross-compare landmark detection results. Forced-aligned boundaries tend to move around significantly as different acoustic models have been used to conduct the alignment. However, we could at least reference this alignment. The alignment in the figure is generated using `tri3b` acoustic model from the default Kaldi recipe. As we can see, vowels and glides really do not return good detection results. Many vowels are detected as glides, and consonants have been interpreted as glides

as well. Even if we do not consider the problem of vowel and glide, the detection result of consonants also needs more improvement, especially near the center of the utterance, around where the phone 'm' and the second 't' is pronounced, where a lot of insertion can be observed. When a number of detection results have been checked, problems with vowel or glide seem to be common issues. As we will discuss in more detail later, studies exist claiming vowel and glide landmarks are harder to detect, and easily confused [69].

A number of future plans are listed below to improve landmark detection accuracy. These attempts do not necessarily depend on each other. However, they are listed in the chronological order in which I plan to explore them. More easy and conventional attempts will be carried out first, while more innovative, yet also challenging, methods will be tried out later. The early attempts have a good chance of improving the detection accuracy, yet they also have foreseeable limitations. The later attempts will be more unconventional, and they might not return improvement at all. However, if found effective, they have great potential.

Applying More Advanced Neural Network Models

The current landmark detection result reported in Chapter 4 is obtained through fully connected neural networks. However, as ASR and other applications with sequential input have proven, models with memory, such as LSTM [125], or the ability to look into wider context, such as TDNN [126], return higher accuracy. For a collection of corpora (including TIMIT, Iban), the lowest error rate ASR, according to Kaldi examples, uses TDNN acoustic model.

While landmark detection is a different task than the phone recognition conducted by ASR, experiments reported in Chapter 4 found that setups that improve phone recognition accuracy also benefit landmark detection. Therefore, the chance that a landmark detector based on LSMT or TDNN can return higher accuracy is promising. Yet, altering the neural network model is not a major change. Based on experience from ASR, the change will improve accuracy, but improvements are usually bounded to single-digit percentage.

However, this simple approach is not without problems. The reason is that LSTM and

TDNN models usually require higher computational load when compared to fully connected models. This is not a problem for using landmark detection as a task for MTL. Since all computation involving landmarks in the the MTL flow is carried out during training, a more complicated model only means longer training time. This, however, might pose a challenge for frame dropping and re-weighting methods mentioned in Chapter 3. In order for frame-dropping and re-weighting to be carried out during recognition, landmark detection has to be carried out at the same time. If landmark detecting models are too complicated to evaluate, the computational reduction from dropping frames might not make up for landmark detection. This will render the frame-drop strategy pointless. It will be challenging trying to find a balance between a model that is accurate and a model that is cheap to compute.

Training a CTC Model for Acoustic Landmark Detection

CTC [10] has offered a new solution to training statistic models based on neural network when the input is in the form of sequences. The model resulting from the training is not necessarily more accurate, yet it usually simplifies the training procedure. As opposed to the classic training procedure leveraging training labels from forced alignment, which needs to be generated by another already trained acoustic model, CTC training does not require frame-wise labels. This means CTC training does not require forced alignment results, which means that, systematically, it does not need phone boundary information for the training. This opens new possibilities for landmark detector training. The current limitation of landmark detector training is largely imposed by shortage of training data. The unfortunate truth is only very limited corpora such as TIMIT have good human-annotated phone boundaries. TIMIT is not a very large corpus (around 10h). However, larger corpora with more realistic conditions do not have phone boundary annotations. This fact prevents them from being used as training corpus for landmark detection. One might argue that forced alignment results can be used as a replacement for human-annotated phone boundaries, and unfortunately, some attempts have been carried out to leverage forced alignment on TIMIT.

However the study covered in Chapter 3 also found that forced alignment does not return the same performance as human-annotated phone boundaries. When CTC is introduced into landmark detector training, it is possible to train landmark detectors on much larger corpora. However, it is not clear if CTC will find the landmarks, or if what is found by CTC is in fact acoustic landmarks.

Figure 1.2 from [10] illustrates a classic output of a CTC model. Since CTC defines loss based on mismatch in the output and target phone sequence as opposed to `hard` frame-by-frame difference between frame-wise output and label, the training procedure does not penalize phone labels generated outside of the phone duration as far as it is part of a correct sequence. This results in output similar to that presented in Fig 1.2. As we can see, prediction of many phones, 'm' for example, appeared long after the pronunciation of 'm' ended. If applied directly, CTC models have little chance of pointing out where landmarks actually are. One potential solution to this problem is to add an extra criterion during the CTC training procedure to penalize landmark labels predicted at the wrong location. Figure 6.3 illustrates the idea in a rough way. The model will be trained on TIMIT with landmark location loss and other corpora without this restriction. The ideal result of this combination is that extra training data from corpora without human-annotated boundaries can improve the model performance by showing it more variants of phones while the boundary information of TIMIT is good enough to teach to the model to find where the landmark occurs.

Unfortunately, due to the fact that CTC models are more difficult to train and converge slower, it is probably more realistic to relax the restriction on label location, for example to penalize only labels falling far away from landmark location. In this case, it is not clear if the CTC model will present landmarks as they are currently defined. However, if the outcome serves the task in Chapter 3 and 4 well, there is no need to stay with the conventional definition of acoustic landmark.

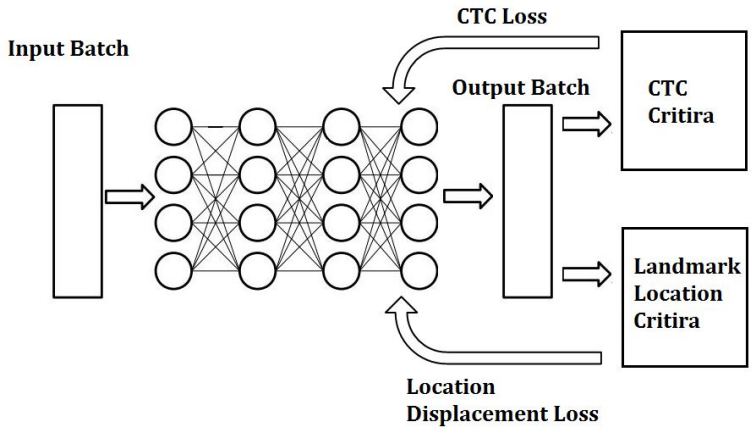


Figure 6.3: Training a CTC-based landmark detector with location information.

6.2.2 Re-defining Acoustic Landmarks

As we have observed in Fig 4.2, as vowel and glide landmarks are defined in their current form, detectors have difficulty finding and locating them. Works such as [69] even claim that vowels do not contain landmarks. Even though acoustic landmarks demonstrate potential, they are only the means of this study; the end is still improving ASR and other speech processing systems. As a result, there is no reason not to re-define landmarks to better serve the need of augmenting ASR. In fact, studies based on the attention model, such as [127], have attempted to re-define the TIMIT phone set and have shown experimental evidence suggesting that a set of phone labels can be found through a data-driven procedure using the attention framework.

CTC models tend to hold back until evidence of a phone is sufficient; this behavior, in many cases, resulted in phone labels being predicted outside the phone boundary. However, if in a context-dependent situation, the location phone labels tend to appear to converge to a fix location, or if there is a mean to restrict it to appear in a desired range, such as the method mentioned in Sec 6.2.1, then can we define these locations as landmarks? One step further, if we supply these locations to a frame-synchronized system, would the latter also benefit from the extra information through methods illustrated in Chapter 3 and 4?

In contrast, if we address head-on the problem of vowel and glide landmarks, what can

we do? A simple change to the current framework is to remove vowel and glide landmarks. However, this change has a negative effect on the usefulness of landmarks as a heuristic since a great portion of phones are made up by vowel and glide. A compensating move might be to augment ASR with phone boundaries as opposed to landmarks. Phone boundaries overlap landmarks if we do not count the vowel and glide landmark. This idea's effectiveness is yet to be checked. Some experiments conducted in the early phase of Chapter 3 hint that they are not as not as informative as landmarks, yet if they can be detected more accurately, the trade-off just might turn out to be in favor of ASR.

6.2.3 Augment CTC Training through Other Means

The study presented in Chapter 5 pre-trains the CTC AM model with landmark augmented label sequence. The AM is then finetuned on a different output layer to ensure the estimated acoustic likelihood by model is consistent with the phone set defined for the language model. However, the model occasionally suffers an illy initialized output layer at the beginning of the finetune procedure and the results are not ideal. This problem is especially significant when the training data is limited. In some cases, the model will not converge in the finetune phase of the training despite converging in the pre-train stage. From this perspective, the pre-train and finetuning strategy is not ideal and other means should also be considered in order to leverage the acoustic landmark information.

The key shortcoming of the pre-train and finetuning strategy is that the transition between the two phases is not necessarily smooth. If the old weights can be leveraged after the transition or the finetuning procedure can start during the pretrain, the negative effect can be minimized. Preliminary attempts have been made during the study, and stacking the output layer of the finetuning network on top of the pre-train network does not seem to return better results. However, methods leveraging MTL frameworks similar to studies presented in Chapter 4 return encouraging results.

When leveraging the MTL framework, the same network is trained simultaneously on

Table 6.1: CTC AM model trained in very under-resource setup.

LER %	70% TIMIT training data	75% TIMIT training data
Baseline	91.84 (not converging)	91.66 (not converging)
Pre-train & finetune	89.77 (not converging)	93.22 (not converging)
MTL	33.8	32.66

both the original phone sequence and acoustic landmark augmented label sequence. The network branches out on the final fully-connected layer. Each layer layer is trained on a separate sequence. The loss of the two final layers is weighted while calculating the gradient for back-propagation. As we can see in Fig 5.5, when the resource used to train AM is reduced below 80% of the TIMIT training set, even with the pre-train and finetuning strategy presented in Chapter 5, the AM has a weak chance of converging. However, preliminary results, illustrated in Tab 6.1, using the MTL framework show that, despite relatively high LER, an AM trained on 70% of the training data can still converge.

The findings presented in Tab 6.1 open new potential for the study presented in Chapter 5 and merit further investigation.

6.2.4 Acoustic Landmark and Attention Models

One of the most meaningful findings of Chapter 3 is that in an input feature frame sequence, some frames are more informative than others. In addition, focusing on these frames benefits an ASR system. In the study, acoustic landmarks were found to indicate the location of these more significant frames. However, this idea that an ASR system can focus on on a subset of the feature sequence is in fact very similar to the core idea of recent studies based on attention [11] and self-attention framework [128]. In the later framework, it is believed that paying different levels of attention to different parts of the input feature sequence creates better prediction results.

Considering that acoustic landmarks have been proven to contain indicative cues, it is very possible that this information might serve as a good heuristic for attention-based AMs.

Past works, such as [127], have already looked into the potential of leveraging attention-based models to conduct AM study from a phonetic perspective. However, these works follow a purely data-driven approach. It is of great importance that audio perception information such as acoustic landmarks might play a positive role in these discoveries.

6.2.5 Verify the Findings on Larger Corpus

Experiments conducted in Chapter 3 and 4 still need to be repeated on larger corpora. The corpora returning the current results are small by today's standard, so they do not lend strong support to the conclusion. More importantly, the result on the current corpus alone is not enough to generalize the conclusions.

For the findings in Chapter 3, this means repeating the experiment on a larger English corpus such as the WSJ.[1] However, this will not be possible without a reasonably reliable landmark detector.

The findings in Chapter 4 should be examined with a larger English corpus and larger second-language corpora in more languages. This might mean conducting the same results on a corpus such as BABEL.[2] Due to the vast difference between languages, there is a good chance that experimental findings will diverge, yet it is still useful to identify a subset of languages that can benefit from landmark detectors trained in English.

[1] `https://catalog.ldc.upenn.edu/ldc93s6a`
[2] `http://mi.eng.cam.ac.uk/~mjfg/BABEL/index.html`

References

[1] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," 2016. [Online]. Available: arXivpreprintarXiv:1610.05256

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim *et al.*, "English conversational telephone speech recognition by humans and machines," in *Interspeech*. ISCA, 2017, pp. 132–136.

[3] E. Marchi, F. Vesperini, F. Eyben, S. Squartini, and B. Schuller, "A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional lstm neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1996–2000.

[4] A. J. Simpson, G. Roma, and M. D. Plumbley, "Deep Karaoke: Extracting vocals from musical mixtures using a convolutional deep neural network," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2015, pp. 429–436.

[5] K.-F. Lee, "Context-independent phonetic hidden Markov models for speaker-independent continuous speech recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 4, pp. 599–609, 1990.

[6] G. Valenzise, L. Gerosa, M. Tagliasacchi, F. Antonacci, and A. Sarti, "Scream and gunshot detection and localization for audio-surveillance systems," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*. IEEE, 2007, pp. 21–26.

[7] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," in *Signal Processing Magazine, IEEE*, vol. 29, no. 6. IEEE, 2012, pp. 82–97.

[8] D. Povey and G. Saon, "Feature and model space speaker adaptation with full covariance gaussians," in *Ninth International Conference on Spoken Language Processing*, 2006.

[9] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE, 2013, pp. 55–59.

[10] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," 2015. [Online]. Available: arXivpreprintarXiv:1507.06947

[11] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, 2015, pp. 577–585.

[12] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, "Deep speech 2: End-to-end speech recognition in English and Mandarin," in *International Conference on Machine Learning*, 2016, pp. 173–182.

[13] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[14] N. Zeghidour, Q. Xu, V. Liptchinsky, N. Usunier, G. Synnaeve, and R. Collobert, "Fully convolutional speech recognition," *CoRR*, vol. abs/1812.06864, 2018. [Online]. Available: http://arxiv.org/abs/1812.06864

[15] H. Von Helmholtz, *On the Sensations of Tone as a Physiological Basis for the Theory of Music*. Longmans, Green, 1912.

[16] P. N. Vassilakis, "Auditory roughness as means of musical expression," *Selected Reports in Ethnomusicology*, vol. 12, 2005.

[17] E. Zwicker and H. Fastl, *Psychoacoustics: Facts and Models*. Springer Science & Business Media, 2013, vol. 22.

[18] K. N. Stevens, "Evidence for the role of acoustic boundaries in the perception of speech sounds," in *Phonetic Linguistics: Essays in Honor of Peter Ladefoged*, V. A. Fromkin, Ed. Orlando, Florida: Academic Press, Cambridge MA USA, 1985, pp. 243–255.

[19] Stevens, Kenneth N, *Acoustic Phonetics*. Cambridge: MIT Press, Cambridge MA USA, 2000, vol. 30.

[20] Z. Wu, C. Valentini-Botinhao, O. Watts, and S. King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *ICASSP*, April 2015, pp. 4460–4464.

[21] M. L. Seltzer and J. Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6965–6969.

[22] D. Chen, B. Mak, C. C. Leung, and S. Sivadas, "Joint acoustic modeling of triphones and trigraphemes by multi-task learning deep neural networks for low-resource speech recognition," in *ICASSP*, 2014, pp. 5592–5596.

[23] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[24] T. Ahmed, M. Uppal, and A. Muhammad, "Improving efficiency and reliability of gunshot detection systems," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 513–517.

[25] L. Gerosa, G. Valenzise, M. Tagliasacchi, F. Antonacci, and A. Sarti, "Scream and gunshot detection in noisy environments," in *Signal Processing Conference, 2007 15th European*. IEEE, 2007, pp. 1216–1220.

[26] C. Clavel, T. Ehrette, and G. Richard, "Events detection for an audio-based surveillance system," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*. IEEE, 2005, pp. 1306–1309.

[27] J. J. Donovan and D. Hussain, "Audio-video tip analysis, storage, and alerting system for safety, security, and business productivity," Aug. 16 2011, US Patent 7,999,847.

[28] S. Moroz, M. Pauli, W. Seisler, D. Burchick, M. Ertern, and E. Heidhausen, "Optical muzzle blast detection and counterfire targeting system and method," Feb. 9 2005, US Patent App. 11/052,921.

[29] X. Zhuang, X. Zhou, M. A. Hasegawa-Johnson, and T. S. Huang, "Real-world acoustic event detection," *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1543–1551, 2010.

[30] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "CNN architectures for large-scale audio classification," 2016. [Online]. Available: arXivpreprintarXiv:1609.09430

[31] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell, "Soundsense: scalable sound sensing for people-centric applications on mobile phones," in *Proceedings of the 7th International Conference on Mobile Systems, Applications, and services*. ACM, 2009, pp. 165–178.

[32] J.-C. Wang, J.-F. Wang, and Y.-S. Weng, "Chip design of MFCC extraction for speech recognition," *INTEGRATION, the VLSI journal*, vol. 32, no. 1, pp. 111–131, 2002.

[33] T. Wu, Y. Yang, Z. Wu, and D. Li, "MASC: a speech corpus in mandarin for emotion analysis and affective speaker recognition," in *Speaker and Language Recognition Workshop, 2006. IEEE Odyssey 2006: The*. IEEE, 2006, pp. 1–5.

[34] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dartaset for audio events," in *IEEE ICASSP*, 2017.

[35] L. H. Arnal, A. Flinker, A. Kleinschmidt, A.-L. Giraud, and D. Poeppel, "Human screams occupy a privileged niche in the communication soundscape," *Current Biology*, vol. 25, no. 15, pp. 2051–2056, 2015.

[36] P. N. Vassilakis and K. Fitz, "Sra: A web-based research tool for spectral and roughness analysis of sound signals," in *Proceedings of the 4th Sound and Music Computing (SMC) Conference*, 2007, pp. 319–325.

[37] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 578–589, 1994.

[38] O. Lartillot, P. Toiviainen, and T. Eerola, "Mirtoolbox," 2008.

[39] W. Aures, "A procedure for calculating auditory roughness," *Acustica*, vol. 58, no. 5, pp. 268–281, 1985.

[40] K. Banks, "The Goertzel algorithm," *Embedded Systems Programming*, vol. 15, no. 9, pp. 34–42, 2002.

[41] A. Dufaux, "Detection and recognition of impulsive sounds signals," *Institute de Microtechnique Neuchatel, Switzerland*, 2001.

[42] R. Bachu, S. Kopparthi, B. Adapa, and B. Barkana, "Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal," in *American Society for Engineering Education (ASEE) Zone Conference Proceedings*, 2008, pp. 1–7.

[43] N. Cressie and H. Whitford, "How to use the two sample t-Test," *Biometrical Journal*, vol. 28, no. 2, pp. 131–148, 1986.

[44] T. Joachims, "Svmlight: Support vector machine," vol. 19, no. 4, 1999. [Online]. Available: SVM-LightSupportVectorMachinehttp://svmlight.joachims.org/ ,UniversityofDortmund

[45] H. Fletcher and W. A. Munson, "Loudness, its definition, measurement and calculation," in *Bell System Technical Journal*, vol. 12, no. 4. Wiley Online Library, 1933, pp. 377–430.

[46] M. Hasegawa-Johnson, J. Baker, S. Borys, K. Chen, E. Coogan, S. Greenberg, A. Juneja, K. Kirchhoff, K. Livescu, S. Mohan *et al.*, "Landmark-based speech recognition: Report of the 2004 Johns Hopkins summer workshop," in *ICASSP*, vol. 1, 2005, p. 1213.

[47] A. Juneja, "Speech recognition based on phonetic features and acoustic landmarks," Ph.D. dissertation, University of Maryland at College Park, College Park, MD, USA, 2004.

[48] A. Jansen and P. Niyogi, "A hierarchical point process model for speech recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 2008, pp. 4093–4096.

[49] J. Iso-Sipilä, "Speech recognition complexity reduction using decimation of cepstral time trajectories," in *Proceedings of 10th IEEE European Signal Processing Conference*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 2000, pp. 1–4.

[50] V. Vanhoucke, M. Devin, and G. Heigold, "Multiframe deep neural networks for acoustic modeling," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 2013, pp. 7582–7585.

[51] I. McGraw, R. Prabhavalkar, R. Alvarez, M. G. Arenas, K. Rao, D. Rybach, O. Alsharif, H. Sak, A. Gruenstein, F. Beaufays *et al.*, "Personalized speech recognition on mobile devices," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 2016, pp. 5955–5959.

[52] K.-F. Lee, *Automatic Speech Recognition: The Development of the SPHINX System*. Springer Science & Business Media, Berlin Germany, 1988, vol. 62.

[53] R. Jakobson, C. G. Fant, and M. Halle, *Preliminaries to Speech Analysis. The distinctive features and their correlates.* MIT Press, Cambridge MA USA, 1951.

[54] N. Chomsky and M. Halle, *The Sound Pattern of English*. MIT Press, Cambridge MA USA, 1968.

[55] K. N. Stevens, S. J. Keyser, and H. Kawasaki, "Toward a phonetic and phonological theory of redundant features," in *Invariance and Variability in Speech Processes*, J. S. Perkell and D. H. Klatt, Eds. Hillsdale, NJ US: Lawrence Erlbaum Associates, New Jersey USA, 1986, pp. 426–463.

[56] K. N. Stevens, "Toward a model for lexical access based on acoustic landmarks and distinctive features," in *The Journal of the Acoustical Society of America*, vol. 111, no. 4. Acoustical Society of America, Melville NY USA, 2002, pp. 1872–1891.

[57] K. Kirchhoff, "Combining articulatory and acoustic informatino for speech recognition in noisy and reverberant environments," in *ICSLP*, 1998, pp. 0873:1–4.

[58] K Kirchhoff, "Robust speech recognition using articulatory information," Ph.D. dissertation, University of Bielefeld, 1999.

[59] K. Kirchhoff, G. A. Finkard, and G. Sagerer, "Combining acoustic and articulatory feature information for robust speech recognition," in *Speech Communication*, vol. 37, no. 3, 2002, pp. 303–319.

[60] K. Livescu, Özgür. Çetin, M. Hasegawa-Johnson, S. King, C. Bartels, N. Borges, A. Kantor, P. Lal, L. Yung, A. Bezman, S. Dawson-Hagerty, B. Woods, J. Frankel, M. Magimai-Doss, and K. Saenko, "Articulatory feature-based methods for acoustic and audio-visual speech recognition: 2006 JHU summer workshop final report," in *Proceedings of IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP)*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 2007, pp. 621–624.

[61] F. Metze, "Articulatory features for conversational speech recognition," Ph.D. dissertation, Fakultät für Informatik der Universität Karlsruhe (TH), Karlsruhe, Germany, December 2005.

[62] A. B. Naess, K. Livescu, and R. Prabhavalkar, "Articulatory feature classification using nearest neighbors," in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2011, pp. 2301–2304.

[63] C.-H. Lee, M. A. Clements, S. Dusan, E. Fosler-Lussier, K. Johnson, B.-H. Juang, and L. R. Rabiner, "An overview on automatic speech attribute transcription (ASAT)." in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2007, pp. 1825–1828.

[64] K. N. Stevens, "The quantal nature of speech: Evidence from articulatory-acoustic data," in *Human Communication*. McGraw-Hill, 1972, pp. 51–66.

[65] S. A. Liu, "Landmark detection for distinctive feature-based speech recognition," in *The Journal of the Acoustical Society of America*, vol. 100, no. 5, 1996, pp. 3417–3430.

[66] S. Furui, "On the role of spectral transition for speech perception," in *The Journal of the Acoustical Society of America*, vol. 80, no. 4. Acoustical Society of America, Melville NY USA, 1986, pp. 1016–1025.

[67] R. N. Ohde, "The developmental role of acoustic boundaries in speech perception," in *The Journal of the Acoustical Society of America*, vol. 96, no. 5. Acoustical Society of America, Melville NY USA, 1994, pp. 3307–3307.

[68] M. Hasegawa-Johnson, "Time-frequency distribution of partial phonetic information measured using mutual information," in *ICSLP*, 2000, pp. 133–136.

[69] S. M. Lulich, "Subglottal resonances and distinctive features," in *Journal of Phonetics*, vol. 38, no. 1, 2010, pp. 20–32.

[70] S. Wang, S. M. Lulich, and A. Alwan, "Automatic detection of the second subglottal resonance and its application to speaker normalization," in *The Journal of the Acoustical Society of America*, vol. 126, no. 6. Acoustical Society of America, Melville NY USA, 2009, pp. 3268–3277.

[71] X. Kong, X. Yang, M. Hasegawa-Johnson, J.-Y. Choi, and S. Shattuck-Hufnagel, "Landmark-based consonant voicing detection on multilingual corpora," 2016. [Online]. Available: arXivpreprintarXiv:1611.03533

[72] A. W. Howitt, "Vowel landmark detection," in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2000, pp. 628–631.

[73] J.-Y. Choi, "Detection of consonant voicing: A module for a hierarchical speech recognition system," in *The Journal of the Acoustical Society of America*, vol. 106, no. 4. Acoustical Society of America, Melville NY USA, 1999, p. 2274.

[74] J.-J. Lee and J.-Y. Choi, "Detection of obstruent consonant landmark for knowledge based speech recognition system," in *Acoustics '08*, Paris, 2008, pp. 2417–2421.

[75] S. Lee and J.-Y. Choi, "Vowel place detection for a knowledge-based speech recognition system," in *Acoustics '08*, Paris, 2008, pp. 2430–2433.

[76] J.-W. Lee, J.-Y. Choi, and H.-G. Kang, "Classification of fricatives using feature extrapolation of acoustic-phonetic features in telephone speech," in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2011, pp. 1261–1264.

[77] Jung-Won Lee and Jeung-Yoon Choi and Hong-Goo Kang, "Classification of stop place in consonant-vowel contexts using feature extrapolation of acoustic-phonetic features in telephone speech," in *The Journal of the Acoustical Society of America*, vol. 131, no. 2. Acoustical Society of America, Melville NY USA, 2012, pp. 1536–1546.

[78] P. Niyogi, C. Burges, and P. Ramesh, "Distinctive feature detection using support vector machines," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 1999, pp. 425–428.

[79] S. Borys, "An SVM front-end landmark speech recognition system," Master's thesis, University of Illinois at Urbana-Champaign, Urbana, IL, USA, 2008.

[80] R. Chitturi and M. Hasegawa-Johnson, "Novel time domain multi-class svms for landmark detection." in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2006, pp. 2354–2357.

[81] Z. Xie and P. Niyogi, "Robust acoustic-based syllable detection," in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2006, p. 1327.

[82] K. Qian, Y. Zhang, and M. Hasegawa-Johnson, "Application of local binary patterns for SVM based stop consonant detection," in *Proc. Speech Prosody*. International Speech and Communication Association (ISCA), Baixas France, 2016, pp. 1114–1118.

[83] J. S. Garofalo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "The DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM," in *Linguistic Data Consortium*, 1993.

[84] K. N. Stevens, S. Y. Manuel, S. Shattuck-Hufnagel, and S. Liu, "Implementation of a model for lexical access based on features," in *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, vol. 1. Banff, Alberta: International Speech and Communication Association (ISCA), Baixas France, 1992, pp. 499–502.

[85] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proceedings of Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2013, pp. 2345–2349.

[86] T. F. Quatieri, *Discrete-Time Speech Signal Processing: Principles and Practice*. Pearson Education, Upper Saddle River NJ USA, 2008.

[87] S. Furui, "Cepstral analysis technique for automatic speaker verification," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 2. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 1981, pp. 254–272.

[88] H. Hermansky, "Trap-tandem: Data-driven extraction of temporal features from speech," in *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*. Institute of Electrical and Electronics Engineers (IEEE), 2003, pp. 255–260.

[89] S. E. Öhman, "Coarticulation in VCV utterances: Spectrographic measurements," *The Journal of the Acoustical Society of America*, vol. 39, no. 1, pp. 151–168, 1966.

[90] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Institute of Electrical and Electronics Engineers (IEEE), Piscataway New Jersey US, 1989, pp. 532–535.

[91] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling." in *Proceedings of 15th Annual Conference of the International Speech Communication Association (INTERSPEECH)*. International Speech and Communication Association (ISCA), Baixas France, 2014, pp. 338–342.

[92] S. Furui, "On the role of spectral transition for speech perception," *Journal of the Acoustical Society of America*, vol. 80, no. 4, pp. 1016–1025, 1983.

[93] B. Delgutte and N. Y. Kiang, "Speech coding in the auditory nerve: IV. Sounds with consonant-like dynamic characteristics," *J. Acoustical Society of America*, vol. 75, pp. 897–907, 1984.

[94] P. Jyothi and M. Hasegawa-Johnson, "Acquiring speech transcriptions using mismatched crowdsourcing," in *Proc. AAAI*, 2015, pp. 1263–1269.

[95] A. Juneja and C. Espy-Wilson, "A novel probabilistic framework for event-based speech recognition," *J. Acoustical Society of America*, vol. 114, no. 4(A), p. 2395, 2003.

[96] D. He, B. P. P. Lim, X. Yang, M. Hasegawa-Johnson, and D. Chen, "Selecting frames for automatic speech recognition based on acoustic landmarks," *Journal of the Acoustical Society of America*, vol. 141, no. 5, pp. 3468–3468, 2017.

[97] D. He, B. P. Lim, X. Yang, M. Hasegawa-Johnson, and D. Chen, "Acoustic landmarks contain more information about the phone string than other frames for automatic speech recognition with deep neural network acoustic model," *Journal of the Acoustical Society of America*, vol. 143, no. 6, pp. 3207–3219, 2018.

[98] R. Caruana, *Multitask Learning*. Boston, MA: Springer US, 1998, pp. 95–133. [Online]. Available: https://doi.org/10.1007/978-1-4615-5529-2_5

[99] X. Yang, K. Audhkhasi, A. Rosenberg, S. Thomas, B. Ramabhadran, and M. Hasegawa-Johnson, "Joint modeling of accents and acoustics for multi-accent speech recognition," 2018. [Online]. Available: arXivpreprintarXiv:1802.02656

[100] A. R. Barron, "Approximation and estimation bounds for artificial neural networks," *J. Machine Learning*, vol. 14, pp. 115–133, 1994.

[101] S. Samson, L. Besacier, B. Lecouteux, and M. Dyab, "Using resources from a closely-related language to develop asr for a very under-resourced language: A case study for iban," in *Interspeech 2015*, Dresden, Germany, Sep. 2015. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01170493

[102] A. Stolcke, "SRILM–An extensible language modeling toolkit," in *ICSLP*, 2002.

[103] P. Boersma and D. Weenink, "PRAAT, a system for doing phonetics by computer, version 3.4," *Institute of Phonetic Sciences of the University of Amsterdam, Report*, vol. 132, p. 182, 1996.

[104] X. Yang, A. Loukina, and K. Evanini, "Machine learning approaches to improving pronunciation error detection on an imbalanced corpus," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 300–305.

[105] A.-r. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14–22, 2012.

[106] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[107] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *International Conference on Machine Learning*. ACM, 2006, pp. 369–376.

[108] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *ASRU*. IEEE, 2015, pp. 167–174.

[109] Y. Miao, M. Gowayyed, X. Na, T. Ko, F. Metze, and A. Waibel, "An empirical exploration of CTC acoustic models," in *ICASSP*. IEEE, 2016, pp. 2623–2627.

[110] H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, "Learning acoustic frame labeling for speech recognition with recurrent neural networks," in *ICASSP*. IEEE, 2015, pp. 4280–4284.

[111] C. Niu, J. Zhang, X. Yang, and Y. Xie, "A study on landmark detection based on CTC and its application to pronunciation error detection," in *APSIPA ASC*. IEEE, 2017, pp. 636–640.

[112] D. He, B. P. Lim, X. Yang, M. Hasegawa-Johnson, and D. Chen, "Improved ASR for under-resourced languages through multi-task learning with acoustic landmarks," in *Interspeech*. ISCA, 2018, pp. 2618–2622.

[113] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *the Workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[114] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International Conference on Machine Learning*. ACM, 2014, pp. 1764–1772.

[115] Y. Wang, X. Deng, S. Pu, and Z. Huang, "Residual convolutional CTC networks for automatic speech recognition," 2017. [Online]. Available: arXivpreprintarXiv: 1702.07793

[116] X. Kong, X. Yang, M. Hasegawa-Johnson, J.-Y. Choi, and S. Shattuck-Hufnagel, "Landmark-based consonant voicing detection on multilingual corpora," *Journal of the Acoustical Society of America*, vol. 141, no. 5, pp. 3468–3468, 2017.

[117] K. N. Stevens, S. Y. Manuel, S. Shattuck-Hufnagel, and S. Liu, "Implementation of a model for lexical access based on features," in *Second International Conference on Spoken Language Processing*. ISCA, 1992, pp. 499–502.

[118] K. N. Stevens, "Evidence for the role of acoustic boundaries in the perception of speech sounds," *The Journal of the Acoustical Society of America*, vol. 69, no. S1, pp. S116–S116, 1981.

[119] J. J. McCarthy, "Feature geometry and dependency: A review," *Phonetica*, vol. 45, no. 2-4, pp. 84–108, 1988.

[120] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.

[121] H. A. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Springer Science & Business Media, 2012, vol. 247.

[122] D. He, Z. Cheng, M. Hasegawa-Johnson, and D. Chen, "Using approximated auditory roughness as a pre-filtering feature for human screaming and affective speech AED," in *INTERSPEECH*, 2017, pp. 1914–1918.

[123] X. Zhang, A. Ramachandran, C. Zhuge, D. He, W. Zuo, Z. Cheng, K. Rupnow, and D. Chen, "Machine learning on FPGAs to face the IoT revolution," in *Proceedings of the 36th International Conference on Computer-Aided Design*. IEEE Press, 2017, pp. 819–826.

[124] D. He, X. Yang, B. P. Lim, Y. Liang, M. Hasegawa-Johnson, and D. Chen, "When CTC training meets acoustic landmarks," 2018. [Online]. Available: arXivpreprintarXiv:1811.02063

[125] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (icassp), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649.

[126] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[127] J. Li and M. Hasegawa-Johnson, "A comparable phone set for the timit dataset discovered in clustering of listen, attend and spell," *NIPS 2018 Workshop IRASL*, 2018.

[128] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.